technische universität
dortmund

fakultät für informatik

# Face Recognition Based On Interest Points

Sebastian Stein

Diplomarbeit
Juli 2010

interne berichte
internal reports

Fakultät für Informatik
Technische Universität Dortmund

GUTACHTER:
Prof. Dr.-Ing. Gernot A. Fink
Dr.-Ing. Thomas Plötz

fakultät für
informatik

GERMANY | D-44221 DORTMUND

# Abstract

Face recognition attracts the attention of researchers since more than 35 years and is still an unsolved research problem. This history reflects not only the complexity of this task but also the strong interest in machine recognition systems, which is mainly driven by the wide range of potential application areas.

In this diploma thesis, we design a combined face detection and identification system based on SIFT descriptors.

We adapt the learning procedure of an existing face detection model using an agglomerative clustering approach. For face identification, we propose a new model using SIFT descriptors and the Object Class Invariant (OCI), which is a scale and orientation invariant reference frame. We introduce a new feature matching strategy based on this OCI, which efficiently reduces the number of potential feature matches and false correspondences. This selection strategy does not only incorporate the 2D image position but also the scale and the angle of a feature. We systematically evaluate the impact of various similarity measures on identification performance. The proposed face identification method is evaluated on the large scale FERET face database and shows to be competitive with other face identification methods such as SIFT-Cluster and Local Binary Patterns. Furthermore, the OCI based identification model does not require preprocessing steps such as histogram equalization or rasterization. Labeling the OCI in an image is sufficient to identify faces invariant to scale and in-plane rotation.

For the combination of face detection and identification, we experiment with two different approaches: A loose coupling technique uses OCI information provided by the detector to define a scale and orientation invariant bounding box, based on which features for face identification are selected. We also experiment with a rather tight system combination strategy, which inspects only features used for detection to identify a face.

# Contents

# 1 Introduction

## 1.1 Face Recognition

In computer vision, face recognition attracts the attention of researchers since more than 35 years and is still an unsolved problem [ZCRP03]. This history reflects not only the difficulty and complexity involved in automating this task, but also the strong interest in machine recognition systems, which is mainly driven by the wide range of potential application areas. The problem can be generally stated as follows: Given a still image of a scene or a video frame of a video camera, recognize faces of known individuals. This section highlights some general aspects of face recognition and in particular with respect to computer vision.

### Application Areas

For law enforcement, surveillance, and authentication systems, automated face recognition exhibits several particular characteristics which would make a reliable system based on distinctive facial traits favorable over other biometric methods such as fingerprint or iris recognition:

Analyzing the image of a persons face is rather user-friendly and intuitive. It could be done *on the fly* - without requiring (a great deal of) user interaction, while for methods including iris scans or fingerprints a comparatively small distance between the scan device and the user as well as his or her co-operation are inevitable. This characteristic implies that face recognition systems do not even rely on the participants knowledge. Considering the widespread use of surveillance cameras in and around buildings or even in outside public areas[1] today, such recognition systems could be deployed as an upgrade to already existing infrastructure. It should be mentioned that new European passports

[1] i.e. C.C.T.V.

are equipped with face images and fingerprints of their holders, which renders a centralized database of *face templates* for public applications more or less unnecessary.

Besides security related scenarios, applications in other research areas would greatly benefit from automatic face recognition. In the fields of virtual reality, ubiquitous computing, human-computer-interaction/ human-robot-interaction and entertainment, numerous useful and creative applications are imaginable.

### Problem Complexity

In order to describe the difficulties involved in solving the face recognition problem in more detail, the following subdivision of this task will be shown to be useful:

1. Given an image, detect and localize all faces in the scene.

2. For each detected face, resolve the name of the corresponding person or mark it as unknown.

At first, we want to detect an unknown number of face occurrences and determine their positions in a given image. Therefore, we need a general idea of the appearance of a face independent of the underlying person. In the second subtask on the other hand, the goal is to distinguish faces belonging to different people, while recognizing faces corresponding to the same person as such. Hence we need a model for the appearance of a face corresponding to a particular person. Although the two subproblems deal with faces, intuitively their solutions and thus the underlying models might be quite different. In terms of pattern recognition and machine learning, the former task is a two class classification problem, whereas the latter one is a multiple class problem each of them being a member of the more general face category determined by the first subtask. In fact, these two distinct problems are addressed separately in the literature, and only a few particular methods are applied to both tasks, where one of them can be solved as a by-product of solving the other.

In the remainder of this thesis we call the task of detecting and localizing faces in a scene image *face detection*, and the task of finding the corresponding name to a given face image *face identification*. The combination of face detection and identification is called *face recognition*. A scenario, in which we wish to

determine whether a face image belongs to one particular person is called *face authentication*.

Existing techniques can be broadly grouped into two categories, based on the *features* they extract, or more generally speaking, on the way they use the information provided by the input image. **Holistic** methods treat the image as a single high-dimensional vector of intensities - one dimension for each pixel assuming gray scale images. Each image of the same size lies within this *image space*. They proceed by reducing the dimensionality of the image space in order to create a domain specific subspace (e.g. *face space*). Projecting a new image onto this subspace, reveals characteristic information about the image content. Well-known examples are *Principle Component Analysis* (PCA) and *Linear Discriminant Analysis* (LDA), which - in the context of face recognition - are termed *Eigenfaces* and *Fisherfaces*, respectively. **Local/structural** approaches extract characteristic *local image features* focussing on specific regions of interest (e.g. eyes, nose and mouth), and define a geometric relationship between these parts of a face. One prominent representative for this category is *Elastic Bunch Graph Matching*. An outline of these and other methods is given in Chapter 2.

Although many different approaches to solving the face recognition problem emerged during the past decades, unfortunately none of them works in an unconstrained setting with *acceptable*[2] accuracy. This is especially due to their lack of robustness against a number of variations occuring in realistic scenarios:

Non-rigid geometric deformations arise naturally with a change of facial expression. The geometric relationships between parts of a face are not fixed and hence more difficult to model than in the case of rigid objects. However, *loose* geometric restrictions may be defined, which accept small variations. Changes in the pose of a face relative to camera viewpoint, appearing as rotations and translations in the image plane as well as in depth, result in a significantly different face appearance. Additionally, the face texture and shape changes with aging. Hence the face appearance of a specific person is subject to large variations.

Partial occlusions in crowded scenes, through the use of eyeglasses or hats and with facial hair escalate the incompleteness of face information provided by an image. Therefore, *parts based* face models using collections of local features

[2]assuming an accuracy greater or equal to 99% is considered as acceptable

are particularly useful. Illumination varies with a change of camera position, number, position(s) and kind(s) of light source(s) and considering the sun, illumination changes occur solely with time (especially outdoors). Clutter in the background complicates the separation of the face from the background (this process is referred to as *segmentation*).

The union of these appearance variations is particularly challenging for the face identification problem, in which members of different classes (i.e. faces of different people) are rather similar. Hence, there is a strong need for models which are able to cope with the large appearance variations of a person's face on one side, and distinguish faces of different people on the other side. However, existing systems may already be used in environments less relevant to security. A promising approach to using current face recognition techniques in security related scenarios is to combine them with other authentication techniques such as fingerprint [RK+07].

## 1.2 Interest Points

Interest points, also known as key points or corner points, are mathematically well-defined image positions, whose surrounding image regions hold locally characteristic information for the scene in an image. Interest point descriptors, which are the *features* extracted at these points of interest, are designed to be *stable* under certain image transformations such as scaling or rotation.

Consider an image showing an object on a black background. Then each interest point descriptor extracted from this image *describes* the appearance of a part of this object. Thus, the appearance of the whole object may be represented by the whole set of features extracted from this image (in the following, this set of features is called *template*). Therefore, methods based on interest point for object recognition belong to the *local/ structural approaches* as defined above. Now consider an image showing the same object in a more complex scene. Some descriptors extracted from this scene image probably correspond to the object in question. Depending on the *robustness* of an interest point descriptor, it is possible to find feature correspondences - called *feature matches* - between the image of the single object and the scene image. These feature correspondences form the basis of very powerful applications. Interest point methods are applied to 3D reconstruction, general object detection, object tracking and other prob-

lems of computer vision, and showed exceptional performance. First attempts to face detection and face identification suggest a similar potential in this domain.

Generally, an interest point method includes three modules: An *extraction method* describing how to find key points in an image automatically, a *description method* defining which information is extracted from the local image region around a key point, and a *matching strategy*, which describes how to find feature correspondences between a pair of images.

In order to successfully recognize an object (e.g. a face) in a scene image, it is not necessary to find feature correspondences for *all* template features. Therefore, these methods perform well on occluded objects. As interest point descriptors are more or less robust against certain image transformations, objects may also be recognized in images taken under different illumination conditions and partially from different perspectives. Furthermore, an object appearing bigger or smaller in a new image or with a different orientation may be recognized successfully. However, this *invariance* to certain viewing conditions strongly depends on the particular interest point method in use. In Section 2.3 we will introduce some of these methods and approaches to object and face recognition based on these methods.

The positive characteristics of methods based on interest points discussed above, as well as the evaluation results of first interest point based approaches to face recognition, motivate to further explore the potential of these methods.

## 1.3 Goals

The main goal of this diploma thesis is to design and implement a combined face detection and identification system exclusively based on interest point methods. This goal will be approached in three steps:

At first, we adapt an existing face detection model using SIFT descriptors. Then, we will carefully design a new face identification model, which facilitates a seamless integration of both models to a fully functional face recognition system. In this context, we will systematically investigate the impact of various similarity measures on identification performance, in order to determine the best similarity measure for this identification system. The final step will be to define a system combination technique.

All of these models - the detection model, the identification model and the

combined recognition model - will be evaluated under various viewing conditions, in order to determine particular strengths and weaknesses of our models.

The remainder of this diploma thesis is organized as follows: Chapter 2 reviews related work on face detection, face identification and particularly interest point based methods for object detection, face detection and face identification. Chapter 3 describes our adaptation of the detection model, the new identification model and two types of system combination techniques. In Chapter 4 evaluation results on the large scale FERET face database are presented and discussed. Chapter 5 concludes and outlines directions for future work.

# 2 Related Work

This chapter introduces a few representative examples of approaches to *face detection* and *face recognition* and discusses their relative advantages and drawbacks. As we have seen in Section 1.1, all of these methods can be generally grouped into *holistic* and *local/structural* approaches. Though keeping this criterion to some extent, this chapter emphasizes a categorization of proposed solutions based on the problem they address. In the special case, where a technique relates to multiple problems, it is introduced in the context of the first, and we state explicitly how it is used for solving the other one. Underlining the importance of methods based on interest points for this thesis, a dedicated subsection will review these approaches in detail. In this context, a review of object detection methods is also provided, as most of the techniques have initially been applied to general objects and many of them have not been evaluated in the face recognition domain.

## 2.1 Face Detection

Face detection is the task of detecting an unknown number of faces in an image and provide a description of their respective locations. Generally, the location of a face is defined by a bounding box around the face region. In this section we introduce a few existing approaches and we refer the interested reader to [Hje01] for a more detailed survey.

### 2.1.1 Holistic Approaches

Holistic methods for face detection intend to decide, whether the inspected image region represents a face or not. Since the input to a face detector is a scene image, which usually contains more than just a single face (e.g. objects in the background, other parts of the body or other faces), holistic face detection

is performed on a (possibly high) number of *windows/ sub-regions* of the image. For example, we could define a fixed width and height of a face and inspect for each pixel in the scene image a window with the predefined size centered at this pixel. For each of these regions, the detector decides whether this particular region represents a face or not. This way, we can localize a face and describe the face location by the pixel position of the center of the inspected region, or by the rectangular region itself.

The problem of deciding whether a region represents a face is a standard two-class classification problem. The image region is simply interpreted as a single high dimensional vector, each dimension representing the intensity of a particular pixel. The whole space of images is called *image space*. Several standard techniques, such as neural networks [RBK98], support vector machines [OFG97] and principle component analysis [SK87] have been applied to describe the subspace of images representing faces - the so called *face space*.

In this section we introduce the *Eigenface* approach, which is based on principle component analysis [SK87]. The Eigenface approach is a rather simple representative of holistic methods.

### Eigenfaces

Interpreting images as vectors enables us to perform mathematical operations on whole images. Considering a set of face images $\{I_i\}, i = 1, \cdots, N$, we can compute the *average face* as

$$\mu_{face} = \frac{1}{N} \sum_{i=1}^{N} I_i \tag{2.1}$$

For each sample face $I_i$, we can estimate the difference $D_i$ to the average face $\mu_{face}$

$$D_i = I_i - \mu_{face}, \tag{2.2}$$

which allows us to describe the covariance matrix of the face distribution in image space: Using the difference matrix $D = (D_1, \cdots, D_N)$, the covariance matrix is is defined as

$$Cov_{face} = DD^T, \tag{2.3}$$

where $T$ denotes matrix transposition. We compute the eigenvectors of this covariance matrix, which have the same dimensionality as the sample face images. Therefore, these eigenvectors may be interpreted as images and are called Eigenfaces. The eigenvalue describes the variance of the distribution in the direction of the corresponding eigenvector. The face space may therefore be sufficiently described by the Eigenfaces $E_j$ corresponding to the top $M$ eigenvalues; the *principle components* of the distribution. This set of Eigenfaces $E_j, j = 1, \cdots M$ forms the face model.

**Detection:** A test image $I_{test}$ may - after subtracting the average face $\mu_{face}$ - be represented as a linear combination of Eigenfaces $E_j$ and a residual error $R$:

$$D_{test} = \sum_{j=1}^{M} w_j \cdot E_j + R \tag{2.4}$$

This remainder $R$ accounts for the fact that we use only the top $M$ Eigenfaces to model the face space. The squared length of the vector $R$ is called *distance from face space* (DFFS) and may be interpreted as the *non-faceness* of a test image. Therefore, the decision of whether an image represents a face is performed by thresholding this non-faceness:

$$face = (R^2 < T), \tag{2.5}$$

where $T$ is the threshold separating faces from non-face images. All images, which may be represented as a linear combination of Eigenfaces and have a sufficiently small residual error are considered as showing a face, while all other images are *rejected*.

**Identification:** The face model introduced above can also be efficiently used for face identification. In this case we focus on the weights $w_i$ in Equation (2.4) rather than the residual error $R$: A face image $I_i$ of a subject $i$ in a face space defined by Eigenfaces $E_j, j = 1, \ldots M, E = (E_1, \cdots, E_M)$ is a point in this $M$-dimensional space. This point $W_i$ is calculated as follows:

$$W_i = E^t D_i, \tag{2.6}$$

which is simply the projection of $D_i$ onto the face space defined by $E$. This $M$-dimensional vector $W_i$ is used to represent the face of a subject as a *template* in the database. For a face image $I_{test}$, which is intended to be identified, the corresponding weight vector $W_{test}$ is calculated using Equation (2.6). The similarity between a test image represented by $W_{test}$ and a known subject represented by $W_i$ is defined as the Euclidean distance between these vectors. The face of subject $i$ whose representation $W_i$ has the minimum distance to $W_{test}$ is considered as being present in the test image.

Although this method is very efficient in the detection/ identification stage in terms of computation time, the approach has several drawbacks: Constructing the face space demands a very large number of training images and a lot of computation time. Furthermore, all face images have to be of the same size and frontal. The accuracy of a detection and identification system using Eigenfaces strongly depends on the number $M$ of principle components used to define the face space. Belhumeur et al. show in [BHK97] that the Eigenface method is particularly sensitive to illumination variations, which can be avoided to some extent by removing the first three principle components (i.e. the eigenvectors with highest eigenvalues). Detection based on Eigenfaces is rather slow considering the large number of possible subregions of a scene image.

### 2.1.2 Local Features for Face Detection

The most prominent approach to face detection using local features has been proposed by Viola and Jones in [VJ01]. Their method inspects subregions of the input image at different positions and scales. Thus, the detector is able to detect faces of different sizes and decides for each subregion individually whether there is a face present or not. The features used are combinations of simple box filters, which are fast to compute on the integral image. The integral image $I_\Sigma$ is defined as $I_\Sigma(x,y) = \sum_{i=0}^{i<x} \sum_{j=0}^{j<y} I(x,y)$, where $I(x,y)$ denotes the input image. Using this representation of an image, a box filter can be computed in constant time, independent of the size of the rectangular regions (boxes) and the scale at which a window is evaluated.

Each such feature $f_i$ in combination with a threshold $t_i$ and a parity $p_i$ constitutes a *weak classifier*: A window is accepted as a potential face if

$$pot\_face = (p_i f_i < p_i t_i) \tag{2.7}$$

is true. The parity $p_i$ is either 1 or -1 and indicates the direction of the inequality sign. Multiple such weak classifiers are combined to form a *strong classifier*. Several strong classifiers are arranged in a so called *cascade*. A window is evaluated using the first strong classifier. If the window is classified as a potential face, it is evaluated using the next classifier. Otherwise it is discarded. This detector design using a series of classifiers allows to rapidly discard subregions of the input image which do not show a face. A face region is only accepted as such, if it passes the whole series of classifiers. Therefore, the cascade is designed to reject most non-face regions based on a very small number of simple features. For the construction of strong classifiers, the authors use the iterative AdaBoost algorithm proposed by Freund and Schapire in [FS95]. Based on a set of face and background training images and a set of features, this algorithm selects the feature, which has the lowest misclassification rate when used as a weak classifier (see Equation (2.7)). In each iteration $t = 1, \ldots, T$, all remaining features are evaluated on the subset of training images which have not yet been correctly classified and the best feature $f_t$ is added to the strong classifier. The algorithm terminates, when a predefined classifier performance criterion is reached. Once the strong classifier in one stage of the cascade is trained, the AdaBoost algorithm is re-initialized with the remaining set of features, all face training images and the subset of background training images, which have not been rejected by any classifier in an earlier stage of the cascade. As a result of this approach, classifiers in early stages are rather simple (i.e. containing a lower number of less complex features), as the whole set of face images may rather easily be separated from the whole set of background images. In later stages of the cascade however, background images which have not yet been rejected are more similar to faces, and therefore classifiers in late stages contain a higher number of more complex features.

The authors report results of a face detector containing 36 stages and a total of 6000 features on the MIT-CMU database[1]. This detector achieves 88% recall at 93% precision. Unfortunately the authors do not state under which conditions a detected face is considered as being correctly localized. In addition to this high accuracy, the detector demands a small amount of computation time. The authors state that the algorithm processes an image of (384 x 288) pixels at a framerate of about 15 Hz on a 700 Mhz Pentium III processor. Note that

[1] Available at: http://vasc.ri.cmu.edu//idb/html/face/frontal_images/index.html

similarly to the Eigenface approach, this method for face detection is single view point in nature. All faces are required to be upright and frontal. The effects of different viewing conditions such as illumination variations and facial expressions on detection performance have not been systematically evaluated.

## 2.2 Face Identification

Considering the long history of research conducted on face identification, it is simply impossible to give a comprehensive overview of existing methods and describe them in an understandable way in the context of this thesis. We therefore concentrate on a few representative examples and refer the interested reader to [ZCRP03] for a more general and extensive introduction.

### 2.2.1 Holistic Approaches

The most basic holistic method (Eigenfaces) for face identification has already been introduced in the context of face detection in the previous section. This approach reduces the dimensionality of the whole *image space* to a much lower dimensional *face space*, while preserving the maximum variance of face images, by choosing the eigenvectors with the largest eigenvalues of the covariance matrix of the distribution of face images. Although this approach is optimal with respect to dimensionality reduction, it is not the most appropriate technique to construct a *face space* which is intended to be used to distinguish faces of different individuals [BHK97]. This is due to the fact that this method does not only preserve the appearance variance *between* faces of different individuals, but also variation due to changing illumination conditions and expression variation of the *same* face. As a result, different images of the same face are not necessarily closer to each other in this feature space than images of different faces. Thus, one may define a different method for constructing the *face space*, incorporating the information of whether variance is caused by face images of different subjects or images of the same face under different viewing conditions.

This idea has been followed by Belhumeur et al., who propose in [BHK97] the so called *Fisherfaces*. The name is derived from the underlying general classification technique called Fisher's Linear Discriminant (FLD), which has initially been applied to taxonomic classification in [Fis36].

Let the set of training images for a single subject be $X_i = \{I_j\}$ and the whole set of training images be $X = X_i$, $i = 1, \cdots, C$, containing $N$ images in total. For each subject $i$, we can estimate the mean appearance $\mu_i$ and the covariance matrix $Cov_i$ using Equations (2.1), (2.2) and (2.3). Similarly we estimate the average face $\mu$ from the whole training set.

Then the *within class scatter* is defined as

$$S_W = \sum_{i=1}^{C} Cov_i \tag{2.8}$$

and the *between class scatter* is defined as

$$S_B = \sum_{i=1}^{C} |X_i| \cdot (\mu_i - \mu)(\mu_i - \mu)^T, \tag{2.9}$$

where $|X_i|$ denotes the number of sample images for subject $i$. Using these scatter matrices, the basis vectors for the face space $F : (F_1, \cdots, F_m)$, which we denoted $E : (E_1, \cdots E_m)$ in the context of Eigenfaces, are determined as to maximize the ratio of between class scatter to within class scatter:

$$F_{opt} = argmax_F \frac{|F^T S_B F|}{|F^T S_W F|}, \tag{2.10}$$

where $|F|$ denotes the determinant of the matrix $F$. The solution to this problem is given by

$$S_B F = S_W F A, \tag{2.11}$$

where $A$ is a diagonal matrix of the corresponding *generalized* eigenvalues. The $M$ vectors $F_i$ with largest generalized eigenvalue are the *Fisherfaces*. In order to find the solution to Equation (2.11), we need to compute the inverse of $S_W$. This matrix is singular if the number of pixels in the image (the dimensionality of the image space) is higher than $N - C$ (the number of training images minus the number of subjects), which is generally the case. Therefore, the authors propose to reduce the dimensionality of the image space using Principle Component Analysis (as described in the previous section) and apply the proposed method to images in this reduced image space.

Evaluation results for this approach are presented in the context of our evaluation in Chapter 4.3. Like Eigenfaces, Fisherfaces need a large number of training

images. Furthermore, this approach requires multiple images per person. Another commonality of these approaches is that they assume faces are aligned, of the same size, upright and frontal. However, identification performance is shown to be very robust under changes in illumination and facial expression. This method is one of the most elaborated holistic approaches to face identification.

### 2.2.2 Local/ Structural Methods

In this section, we introduce *Elastic Bunch Graph Matching* and *Local Binary Patterns*, with which we are going to compare our evaluation results.

#### Elastic Bunch Graph Matching

Elastic Bunch Graphs have been proposed by Wiskott et al. in[WFKvdM97]. This local feature method takes particularly geometrical relationships between features into account.

Features of training images are extracted at predefined, labeled face locations (e.g. eyes, tip of the nose or corners of the mouth). A single feature, called *jet*, consists of 40 complex coefficients obtained by convolving the point in question with 40 Gabor filters [Dau88]. Gabor filter responses are similar to the response of the human cortical receptive field and remove most of the variation caused by small lighting changes and local deformations. Thus, a representation based on Gabor filters is to some extent invariant to these variations.

The jets represent the nodes in a *face graph*. The arcs of this graph are labeled with the distance between the nodes. The whole set of face graphs constructed from training images is combined to a *face bunch graph*. The arcs are set to the average distance of all corresponding arcs in face graphs, and each node of the bunch graph represents a whole *bunch* of jets: "'An eye bunch, for instance, may include jets from closed, open, female and male eyes [...]"'[WFKvdM97]. This bunch graph forms a general face appearance model.

The face of a particular subject is represented by the combination of one jet at each point and the person specific distances between these points. The face graph extracted from a test image is then matched to all face graphs in the database in order to determine the highest similarity. The graph similarity measure takes two factors into account: the jet similarity and the geometrical

distortion. The subject corresponding to the graph in the database with highest similarity to the graph extracted from the test image is considered as being present in the image.

This approach is very flexible and allows to represent faces efficiently. For learning of the face bunch graph, a rather low number of training images compared to Eigenfaces and Fisherfaces is necessary. Furthermore, a subject may be learned for identification based on a single image.

### Local Binary Patterns

The use of Local Binary Patterns (LBP) in the context of face identification has been initially proposed by Ahonen et al. in [AHP04]. LBPs are simple yet descriptive features: The intensities of pixels in the neighborhood of an image position are thresholded by the intensity at the inspected image position. The concatenation of these binary filter responses forms the *local binary pattern*. These features are extracted for all pixels in the image. Let $F(x, y)$ denote the LBP at image position $(x, y)$.

A *circular neighborhood* is defined by a pair $(N, R)$ describing the number of positions $N$ and the distance $R$ of these image locations from the center. Intensities at image positions in between pixels are bilinear interpolated. The total number of possible patterns is $2^N$. This number is reduced by only considering so called *uniform patterns*. "A Local Binary Pattern is called uniform if it contains at most two bitwise transitions from 0 to 1 or vice versa when the binary string is considered circular."[AHP04]. All patterns which do not hold this condition are discarded. Let $P$ be the total number of possible patterns

Local Binary Patterns describe the local structure in the image on a pixel level. In order to describe the image content on a higher level, the whole image is divided into $M$ subregions $R_1, \cdots, R_M$. For each region $j = 1, \cdots, M$, a *histogram* of patterns is computed as follows:

$$H_{ij} = \sum_{(x,y) \in R_j} 1[F(x, y) = i], \tag{2.12}$$

where $1[expression]$ is one if the expression is true and zero otherwise. The histogram entry $H_{ij}$ counts the number of occurrences of pattern $i$ in image region $j$. The histogram of region $j$, $H_j : (H_{1j}, \cdots H_{Pj})$ thus describes the occurrences of all patterns in region $j$. This histogram represents the image

content on the level of an image region. The concatenation of regional histograms $H : (H_1, \cdots H_M)$ forms the whole image representation. Therefore, the content of an image is described by $M \cdot P$ values.

For matching a pair $(A, B)$ of such image descriptions (i.e. a description derived from a test image with a description representing the face of a known individual in the database), the authors define three similarity measures.

Histogram intersection:

$$D(A, B) = \sum_{j=1}^{M} \sum_{i=1}^{P} min(A_{ij}, B_{ij}) \tag{2.13}$$

Log-likelihoood statistic:

$$L(A, B) = -\sum_{j=1}^{M} \sum_{i=1}^{P} A_{ij} \, logB_{ij} \tag{2.14}$$

Chi square statistic:

$$\chi^2(A_{ij}, B_{ij}) = \sum_{j=1}^{M} \sum_{i=1}^{P} \frac{(A_{ij} - B_{ij})^2}{A_{ij} + B_{ij}} \tag{2.15}$$

The authors evaluate the proposed method on the CSU Face Identification Evaluation System [BBTD03] using different neighborhood sizes and different numbers of subregions, with the result that a neighborhood $(8, 2)$ with a region size of 18x21 pixels on 130x150 pixel images may be the best trade-off between descriptor length and identification performance. Comparative evaluation of similarity measures failed to identified one which performs better than the others in all experiments. Yet, the $\chi^2$ method performs best in most experiments. The authors further propose to weight the contribution of a subregion in (2.15) by its descriptiveness:

$$\chi^2(A_{ij}, B_{ij}) = \sum_{j=1}^{M} \sum_{i=1}^{P} w_j \frac{(A_{ij} - B_{ij})^2}{A_{ij} + B_{ij}}, \tag{2.16}$$

where $w_j$ denotes the weight for subregion $j$. In order to determine these weights, the individual identification rate of each subregion is evaluated, averaging the result of corresponding subregions on the left and right half of the face. Regions obtaining less than 0.2 true positive rate are discareded. Weights for regions with a higher true positive rate than 0.8 and 0.9 are set to 2.0 and

4.0, respectiveley. All other regions get weight one. Incorporating this weighting scheme significantly ( ¿ 4 %) increases the true positive rate on all evaluation sets. Detailed evaluation results will be presented in the context of the evaluation of our model in Section 4.3.

## 2.3 Methods Based on Interest Points

This section focusses on interest point extraction, description, and matching techniques and introduces some (probabilistic) models for object and face recognition based on these descriptors. First, we examine methods based on interest points in the area of object detection, as these techniques have initially been applied to general objects.

### 2.3.1 Object Detection

#### Interest Point Descriptors

The most popular interest point methods are the *Scale Invariant Feature Transform* (SIFT), proposed by Lowe in [Low04], and *Speeded Up Robust Features* (SURF) introduced in [BTG06]. As these two methods will be relevant for our introduction of face identification techniques in Section 2.3.2, we will describe them here in some detail.

**Scale Invariant Feature Transform** SIFT descriptors are invariant to changes in scale and in-plane orientation as well as to a wide range of other affine transformations of the local image region around the interest point. They are also partially robust to changes in illumination and 3D viewpoint change.

Interest points are localized at extrema of the Difference-of-Gaussian (DoG) scale-space.. Let $I(x, y)$ be the input image and $G(x, y, \sigma)$ a Gaussian with variable standard deviation $\sigma$. The scale-space is defined as the function

$$L(x, y, \sigma) = G(x, y, \sigma) * I(x, y) \tag{2.17}$$

and the Difference-of-Gaussian is defined as

$$D(x, y, \sigma) = L(x, y, k\sigma) - L(x, y, \sigma) \tag{2.18}$$

for two nearby scales. A pixel with maximal or minimal value in its 26-neighborhood across scales is considered as a potential interest point. Positions with low contrast or with a high edge response are removed. A detailed fit to the nearby data locates the keypoint to sub-pixel accuracy and reveals further unstable locations.

In the next stage, a distinctive orientation is assigned to each keypoint using gradient orientations of pixels in the local image region. The final descriptor is extracted relative to this orientation, which makes it invariant to in-plane rotation. The gradient orientations and magnitudes of the 16x16 image region around the keypoint (relative to the assigned orientation) are used to construct a smoothed 4x4 array of orientation histograms with 8 bins each. These 4x4x8 histogram bins form the 128-dimensional SIFT feature vector. Finally this vector is normalized, which makes it to some extent robust to changes in contrast.

In summary, extracting interest points at scale space extrema makes them invariant to changes in object size. Descriptors relative to a distinctive orientation ensures invariance to rotation in the image plane. Using gradient information of the local image region makes the descriptor invariant to a bias in illumination and vector normalization provides some invariance to changes in contrast. Besides this feature vector describing the appearance of the local image region, the feature geometry (i.e. 2D position in the image, scale and angle) is associated with its descriptor, which allows for determining a geometrical transformation of matching features (and of the underlying objects) across images.

Matching SIFT features is performed as follows: Consider a database of object templates, each described by a set of features $\{f\}^{temp_i}$ and a set of test features $\{f\}^{test}$. For each test feature, we identifiy the closest database feature based on Euclidean distance. Let the corresponding object template be $temp_i$. In order to discard unstable feature matches, the minimum distance to features corresponding to any other object template $temp_j, j \neq i$ is compared to the actual minimum distance. If the distance ratio of the closest and the second closest feature match exceeds a certain threshold, the match is considered to be unstable. Hence, this individual feature matching stage identifies unique matches between test features and features of object templates. In order to verify the hypothesis for a specific object template, each feature votes for any object pose which is consistent with its feature match. An object pose is furtherly verified using the Hough transformation.

The robustness of SIFT descriptors to image deformations has been confirmed by Mikolajczyk and Schmid, who examined different local descriptors in their comparative evaluation presented in [MS05].

Ke and Sukthankar use SIFT's method for interest point selection and orientation assignment in [KS04] and propose a different description method for the local image information based on Principle Component Analysis. They use horizontal and vertical gradients of 41x41 pixels around the keypoint location and project this gradient map onto a precomputed 40-dimensional eigenspace. Evaluation shows that this so called PCA-SIFT method outperforms standard SIFT in controlled matching tasks. Matching using PCA-SIFT is much faster due to the lower dimensional feature descriptors. The evaluated settings contained only planar and rigid objects. Considering the large image region used by PCA-SIFT and the lack of robustness of PCA against non-rigid deformations, it is questionable whether this method compares equally well in the face recognition domain, since the smoothed nature of standard SIFT descriptors suggests a higher tolerance to this kind of variations.

**Speeded Up Robust Features** Bay et al. propose another interest point detection and description method [BTG06]. As the name suggests, the main goal was to design a descriptor which is faster to extract and is comparably robust as SIFT features. Rather than approximating the Laplacian with a difference-of-Gaussian as done by the SIFT method, SURF approximates Gaussian second order partial derivatives using box filters. This type of convolution filters is very fast to compute in the integral image, which is defined as $I_\Sigma(x, y) = \sum_{i=0}^{i<x} \sum_{j=0}^{j<y} I(x, y)$. The computation time of box filter responses in the integral image is $O(1)$. While for the Gaussian scale-space used by SIFT it is necessary to iteratively convolve the input image with a Gaussian filter of increasing size and to subsample the image for each octave, the box filters used here are not applied iteratively and thus can be computed in parallel. Furthermore, with this approach it is not necessary to subsample the input image, as we can simply increase the size of the box filters with no additional computational cost.

Interest points are located at local maxima of the determinant of the Hessian matrix. The sign of the Laplacian (i.e. the trace of the Hessian matrix) holds the information whether we detect a dark blob on a bright background or a

bright blob on a dark background. This information is used in the matching stage to only match similar features, which significantly reduces matching time. An orientation is assigned to each keypoint location using Haar wavelet responses in the local image region, which are simply another type of box filters. Let $d_x$ be the Haar wavelet response in horizontal direction and $d_y$ the Haar wavelet response in vertical direction. The final descriptor is extracted based on the Haar wavelet responses in the 20x20 local image region centered at the keypoint location and rotated to the assigned orientation. This region is split up into 4x4 subregions. Each subregion contributes a four component vector $\mathbf{v} = (\sum d_x, \sum d_y, \sum |d_x|, \sum |d_y|)$ to the interest point descriptor, resulting in a 64-dimensional feature vector. Finally this vector is normalized.

The authors propose two other versions of their descriptor. An upright version called U-SURF, which skips the orientation assignment step, resulting in an even lower computation time. Note that this descriptor is no longer invariant to in-plane rotation. The SURF-128 descritpor is an augmented SURF feature vector with enhanced robustness. The sums of $d_x$ and $|d_x|$ are computed separately for the cases $d_y < 0$ and $d_y \geq 0$. Similarly, the sums of $d_y$ and $|d_y|$ are split up, yielding a 128-dimensional feature vector.

Matching as described in [BTG06] is performed similarly to the SIFT matching strategy. As mentioned above, the sign of the Laplacian is used to significanlty reduce potential feature matches. Based on Euclidean distance, the closesd feature match as well as the second closesd feature match arising from a feature of a different object template are computed. Their ratio is compared to a global threshold to remove unstable feature matches. The template object having the most feature correspondences is considered to be detected. The authors did not incorporate any geometrical constraints or verification methods "'as these may hide shortcomings of the basic schemes"'[BTG06] in their comparative evaluation.

Evaluation showed that the detection and extraction of SURF descriptors is about three times faster than detecting and extracting the same number of SIFT descriptors. The authors also compared their descriptors with SIFT in an object recognition task. The database consisted of 216 images of 22 objects of art in a museum. The test set consisted of 116 image taken from the same objects under different lightning conditions and viewing angles. The augmented SURF-128 descriptor performed best (85.7% recognition rate), followed by U-

SURF (83,8%), SURF (82,6%) and SIFT (78,1%). We will adress the question whether this result of a single rigid object recognition task is transferable to the face recognition domain in Section 2.3.2.

### Probabilistic Models for Object Recognition

Several probabilistic models based on constellations of object parts or interest point descriptors have been proposed in the literature (e.g. [Sch99, PL00, FFFP03, MMP04, MLS06]). This section aims to identify common characteristics and to outline a *general* approach.

The use of probabilistic models for object recognition is motivated by two main factors: The appearance of an object in an image is always subject to some variability. Instead of restricting the *allowed* variability of a feature correspondence by a global and arbitrarily defined threshold, probabilistic models offer the possibility to *learn* the appearance variance for each feature individually from data. Secondly, the confidence (or uncertainty) with respect to an object detection can be expressed in an elegant way by probabilistic object models.

Given a set of observed features in a test image $\{f\}^{test}$, we wish to express the probability of an object $o_i$: $P(o_i|\{f\}^{test})$. A generative model, which tries to describe the scene defined by $\{f\}^{test}$ using knowledge about the object $o_i$, can be obtained using Bayes theorem:

$$P(o_i|\{f\}^{test}) = \frac{P(\{f\}^{test}|o_i)\,P(o_i)}{P(\{f\}^{test})} \tag{2.19}$$

The term $P(o_i)$ is the prior probability of the object. As we generally do not expect to detect one object more frequently than another in a multiple object setting, this probability is constant. The denominator $P(\{f\}^{test})$ is a normalization coefficient ensuring that $P(o_i|\{f\}^{test})$ is a valid probability distribution. Hence this term is negligible. The focus of the model lies therefore on the term $P(\{f\}^{test}|o_i)$, which describes the posterior probability of the set of test features given the object $o_i$. This probability distribution needs to be derived from data. The authors of all afore mentioned articles assume conditional feature independence given the object, which significantly reduces the model complexity.

Making this assumption, the probability simplifies to

$$P(\{f\}^{test}|o_i) = \prod_{j}^{J} P(f_j^{test}|o_i), \tag{2.20}$$

the product of (independent) feature probabilities over all test features $f_j^{test}$, $j = 1, \ldots J$. The conditional feature probability given an object takes several aspects into account: The *appearance probability* of a feature describes the likelihood of a model feature having the observed appearance. This probability describes to some extent the quality of a feature match. Taking SIFT descriptors as an example, the appearance probability models the likelihood of a 128-dimensional SIFT descriptor of a model feature having the form of the observed SIFT descriptor. The *geometrical probability* describes the geometrical distribution of a model feature. Given the hypothesis of a specific object being present in the test image at a specific location, the geometrical occurrence probability distribution is used to verify whether the feature location is consistent with the current object location hypothesis. On the other hand, given a match of a test feature with a model feature, the geometrical distribution is used to infer object locations in the image. The third common aspect taken into account is the feature *occurrence probability*. If a feature occurs in many training images for a given object, it is more likely that a matching test feature observed in a new image accounts for this object than for unrelated clutter. These observations lead to the following general formulation for the probability of a feature given an object:

$$P(f_j|o_i) = P_{appearance}(f_j|o_i)P_{geometry}(f_j|o_i)P_{occurrence}(f_j|o_i) \tag{2.21}$$

**Appearance Clustering** The different approaches in the literature mainly differ in their way of obtaining/ identifying feature appearance clusters, which are distinctive for a specific object. Pope and Lowe propose in [PL00] a multi-view object model, which is basically a two tier clustering approach. For each object class, they construct feature graphs for distinct views of an object. In each of these graphs, features which are similar in appearance and geometry are merged to form a model feature. A problem with this approach arises with model features which are similar in appearance and geometry and belong to different view graphs. In this case, the conditional feature independence assumption is invalid,

which artificially increases the probability of an object given the test feature if this situation is ignored.

Mikolajczyk et al. propose in [MLS06] a hierarchical tree structure for appearance clusters. This tree of appearance clusters is constructed from all training features of all training objects using an agglomerative clustering technique. Clustering starts with each training feature as a potential appearance cluster. Iteratively, the two clusters with minimum distance of their respective cluster centers are merged if this distance does not exceed a certain threshold. This threshold is chosen to be relatively small for the clusters which form the leafs of the tree, and is incrementally increased to construct the appearance tree. Each leaf node holds the parameters of the object specific appearance and geometry distributions. This structure definitely accelerates matching time, but on the other hand it is questionable whether this approach allows the identification of appearance clusters which are distinct for object classes and feature geometries. Unfortunately the authors do not mention how they efficiently model the geometrical distributions given an object, which may take by construction an arbitrary form.

**Recognition** Recognition using this kind of generative model is the task of finding an object or a collection of objects which best describes the scene (i.e. the observed features in the test image) or to decide that there is no known object present. An hypothesis is a set of correspondences of observed features to model features. In order to find the *best* hypothesis, we theoretically need to evaluate all possible hypothesis. In an example case (taken from [Sch99]), where we extract 100 test features and we obtain 10 possible correspondences for each test feature, we would have to evaluate $10^{100}$ hypothesis. As this evaluation is infeasible, the authors propose different heuristics for hypothesis construction intending to find a *good* hypothesis with high probability.

C. Schmid constructs 2000 hypothesis and sums the scores obtained for each object class over all hypothesis. Unfortunately she does not state in [Sch99], how these hypotheses are constructed. All she mentions is that these constructed hypotheses have a high probability of resulting from true feature correspondences.

Pope and Lowe prioritize in [PL00] features with high occurrence probability and high geometrical distinctiveness in order to identify some few initial correspondences. These matchings are used to determine an initial model pose in the

test image using a similarity transformation. Based on this model pose, further features are matched and the pose of the model recalculated.

Moreels et al. follow a similar iterative hypothesis construction approach in [MMP04]. They use a greedy approach to construct partial hypothesis (i.e. hypothesis where features are neither assigned to a model feature nor to the background), iteratively extending the *most promising* partial hypothesis.

In summary, there does not seem to exist a single most effective way to obtain a good correspondence set between test features and model features.

The final decision of whether an object is present in the scene is straight forward. If there is no model for the background, an object is considered to be present if its probability in a given hypothesis exceeds a predefined threshold. In the case where a background model exists, an hypothesis can be tested using the Bayesian decision ratio

$$\gamma(o_i) = \frac{P(o_i)}{P(bg)} \frac{P(o_i|\{f\}^{test})}{P(bg|\{f\}^{test})}, \tag{2.22}$$

where $bg$ denotes the background model and $P(o_i)$ and $P(bg)$ are the prior probability of the object $o_i$ and the background. This ratio of prior probability can be set by hand to influence the true versus false detection rate. If $\gamma(o_i)$ is greater than one, the object $o_i$ is considered to be detected in the test image.

### 2.3.2 Face Identification

This section reviews some approaches to using SIFT and SURF feature descriptors for face identification.

#### Face Authentication with SIFT

In [BLGT06], Bicego et al. conduct a first systematic investigation of the applicability of SIFT for face authentication systems. Recall that face authentication is to determine whether a given face image corresponds to one particular person in the database. They compared three different matching schemes:

- Minimum pair distance matching
- Matching features around the eyes and the mouth
- Matching on a regular grid with overlapping subregions

**Minimum pair distance matching:** Let $\{f_i\}^{test}$, $i = 1, \ldots, M$, be the set of features derived from the test image and $\{f_j\}^{temp}$ with $j = 1, \ldots, N$ be the set of features corresponding to a single person specific template from the database. The minimum pair distance measure is defined as

$$D_{MPD}(\{f_i\}^{test}, \{f_j\}^{temp}) = min_{i,j}(d(f_i^{test}, f_j^{temp})), \tag{2.23}$$

where $d(a, b)$ denotes the euclidean distance between two 128-dimensional SIFT descriptors. A match is accepted or rejected based on a threshold $t >^{?} D_{MPD}$.

**Matching features around the eyes and the mouth:** Based on labeled landmarks in the test and template images, the minimum pair distances are computed for features corresponding to the eyes and the mouth separately. The final distance measure is derived from averaging these two measures, resulting in the formula

$$\begin{aligned} D_{EM}(\{f_i\}^{test}, \{f_i\}^{temp}) =& \frac{1}{2} D_{MPD}(\{f_i\}^{test}_{eyes}, \{f_j\}^{temp}_{eyes}) \\ &+ \frac{1}{2} D_{MPD}(\{f_i\}^{test}_{mouth}, \{f_j\}^{temp}_{mouth}), \end{aligned} \tag{2.24}$$

where $\{f\}_{eyes}$ and $\{f\}_{mouth}$ are the subsets of $\{f_i\}^{test}$ and $\{f_j\}^{temp}$ corresponding to the eyes and the mouth, respectively. Note that these subsets or the image regions from which the subsets are built, need to be labeled for all template and test images.

**Matching on a regular grid** In order to perform matching on a regular grid, it has to be assumed that images are aligned. This matching strategy computes distances for all pairs of corresponding subregions $r = 1, \ldots, R$, and averages the result.

$$D_{GRID}(\{f_i\}^{test}, \{f_j\}^{temp}) = \frac{1}{R} \sum_{r=1}^{R} D_{MPD}(\{f_i\}^{test}_{r}, \{f_j\}^{temp}_{r}), \tag{2.25}$$

**Evaluation** These matching strategies have been evaluated on the BANCA[2] database using the matched controlled protocol involving 52 subjects. During preprocessing, all images were aligned and their intensity histograms were normalized. For each person, 5 images have been used for training and 7 images for testing. In this experiment, matching on a regular grid performed best with an equal error rate of 8.43%, followed by matching eyes and mouth (11.54%) and minimum pair distance matching (13.74%). For regular grid matching, the image was divided into $(4_x \times 2_y) = 8$ subregions. The overlapping was set to 25%.

Note that regular grid matching requires some amount of image registration and that matching the eyes and the mouth depends on accurately labeling or automatically localizing areas corresponding to these face features.

### SIFT Cluster

Luo et al. refined the approach of matching corresponding face regions using SIFT in [LMT+07]. They divide the face image into five subregions using the K-means clustering technique [Mac67]: SIFT features are extracted from registered and aligned training images. Five cluster center locations (i.e. 2D coordinates) are initialized with random values within the registered image space. Each training feature is assigned to the cluster whose center is closer to the feature location than all other cluster centers using Euclidean distance. After this assignment phase, all center locations are updated to the mean positions of their assigned features. Hence features may be assigned to different clusters in the next iteration. Therefore, this process is iterated until cluster centers remain unchanged. The resulting subregions in the image are the *basins of attraction* of these clusters. Each extracted feature from a test image is assigned to one of these subregions, and only features belonging to the same subregion are matched. The proposed matching strategy combines a local and a global similarity measure. Using the notation introduced above, the local measure is defined as

$$D_{local}(\{f_i\}^{test}, \{f_j\}^{temp}) = \frac{1}{R}\sum_{r=1}^{R} min_{i,j}(d(f_i^{test}, f_i^{temp})) \cdot w_r, \tag{2.26}$$

[2]Available at http://www.ee.surrey.ac.uk/CVSSP/banca/

where subregions $r$ are weighted by region specific weights $w_r$, which are determined comparably to [AHP04]. The global similarity measure is defined as

$$D_{global}(\{f_i\}^{test}, \{f_j\}^{temp}) = \frac{match(\{f_i\}^{test}, \{f_j\}^{temp})}{|\{f_j\}^{temp}|}, \tag{2.27}$$

where $match(a, b)$ denotes the number of valid matches according to their distance ratios as defined in [Low04], and $|\{x\}|$ denotes the number of elements in the set $\{x\}$. Note that this global similarity depends not only on the number of matches with respect to a particular template, but also on the ambiguity of matches with respect to all other templates stored in the database. These measurements are combined to a final similarity measure by simple multiplication.

$$D_{final} = D_{local} \cdot D_{global} \tag{2.28}$$

The authors evaluated their proposed approach on the FERET and the CAS-PEAL databases, where all images have been normalized. A detailed presentation of their results is given in the context of our evaluation (Chapter 4). To sum up, this approach is superior to the one of Bicego et. al and shows an incomparable robustness with respect to a viewpoint change of 30°.

#### Identification System for a Mobile Robot

Unlike the previous authors who used face databases for evaluation, Cruz et al. test a SIFT based face recognition system in a realistic scenario using a mobile robot [CSM08]. They propose another matching strategy and a refinement scheme using the information of consecutive frames of a video camera. Face detection is accomplished with the boosted cascade of simple features proposed by Viola and Jones [VJ01] - which is not based on interest points. Once a face is detected, this algorithm is further used to detect an eye, from which they infer "based on standard face measures" the position of two other regions, namely the other eye and the region around the nose and the mouth. Faces are added by storing the SIFT features corresponding to these three regions together with the name and the total number of obtained features in a database. For matching, distances of features of corresponding subregions are computed for all faces in the database. Each single feature match is accepted if it meets the distance ratio criterion proposed by Lowe [Low04]. A face-to-face match is considered to be found based on either of two conditions. Let $s_i$ be the number of valid

matches between the test image and a template $i$, $\mathbf{s} = (s_1, \ldots, s_N)$ the similarity vector for all templates, and $n_i$ the total number of features stored in the $i-th$ template. The basic criterion $\frac{s_i}{n_i} > t$, where $t$ is a predefined global threshold, is combined with one of the following:

$$max(\mathbf{s}) - snd(\mathbf{s}) \geq 2 \cdot snd(\mathbf{s}), \text{ or} \tag{2.29}$$

$$max(\mathbf{s}) - avg(\mathbf{s}) \geq 2 \cdot avg(\mathbf{s}) \tag{2.30}$$

In these formulations $max(\mathbf{s})$, $snd(\mathbf{s})$ and $avg(\mathbf{s})$ denote the maximum, second maximum, and average value, respectively, of $s_i \in \mathbf{s}$.

In their refinement strategy based on consecutive frames, the authors follow a Bayesian approach in order to determine the posterior probability of a face $f_i$ given the similarity vector $\mathbf{s}$, and define probabilistic versions of Equations (2.29) and (2.30). Once a face is identified or a maximum number of frames inspected, the probabilities are reset to uniform.

**Evaluation results** The authors conducted tests on Yale's face database[3] as well as in an indoor office environment using a camera resolution of $640 \times 480$. Once a face has been detected, a tracking algorithm ensured to find the same face in the following frames, and a maximum number of 10 frames is used to determine the identity of a detected face. In this scenario, precision and recall varies between 100% of precision with 33% recall and 96.7% precision with 57.3% of recall. While for a service-robot such a low recall might be acceptable, for security systems using a single image to verify the identity of a person it is simply not. Experimentation results on the Yale database with one test image per person show a precision of 50% with a recall of 10%, where only 17 people are stored in the database.

### SURF for Face Recognition

An investigation on the use of SURF features for face recognition has been carried out by Du et al. in [DSHN09]. Their matching strategy and similarity estimation involves a nearest-neighbor approach based on the Euclidean distance between features in test and training images as above. Rather than constructing subregions to restrict feature matching, they regard only features in the template

[3]Available at: http://cvc.yale.edu/projects/yalefaces/yalefaces.html

image, which lie within a specified search window around the image position of the currently inspected test feature. Therefore, they assume that all faces are upright and face images are normalized. In order to verify a potential feature match, a search over the whole template image region is performed to find a second closest feature match. If the distance ratio is smaller than a predefined threshold, a match is considered to be valid. The final similarity measure between a test and a template image is defined as

$$Sim(\{f_i\}^{test}, \{f_j\}^{temp}) = \begin{cases} \frac{DisAvg + RatioAvg}{2} & \text{if } N \geq 10 \\ \frac{DisAvg + RatioAvg}{2} + 1 & \text{otherwise,} \end{cases} \tag{2.31}$$

where $DisAvg$ denotes the average minimum distance between valid feature matches and $RatioAvg$ denotes the average distance ratio between closest and second closest features. The template with *minimal* similarity is considered as the winner. Evaluation has been carried out on the FERET database [PWHR98, PMRR00] using the subsets with expression variations. One image per person is used for training and one for testing. In comparison with SIFT, SURF shows similar performance but is two times faster for matching. The augmented 128-dimensional version of SURF showed slightly better performance than SURF-64 and standard SIFT.

### Comparison of SIFT and SURF Features

Drew et al. compare SIFT and SURF descriptors in [DSC09] using a slightly different approach. Rather than extracting SIFT and SURF descriptors based on interest points, they extract them on a regular grid. From aligned, normalized and cropped images of $(64 \times 64)$ pixels, they extract 1024 descriptors. For matching, an *all-to-all* and a grid-based approach with and without overlapping are compared. Potential matches are considered as valid if they pass the standard distance ratio test. In order to enhance outlier removal, the authors propose a *random-sample-consensus* (RANSAC)[FB81] approach to find homographies between test and training images and remove matched features whose location projections do not fall into a circular region of three pixels radius from the corresponding template feature location. This RANSAC approach is further used to create a combined system of SIFT and SURF features, where all potential matches are merged before homography determination.

Besides the standard SIFT and SURF procedures, which are in-plane rotationally invariant, the authors evaluate upright versions of both, and the 128-dimensional augmented SURF descriptor.

Tests have been carried out on the AR-Face [4] database as well as on CMU-PIE [SBB02]. All images are rotated, so that the eye-centers lie on the same row. Images are further cropped and scaled to 64 pixels in each direction. Tests on the AR database use seven images per person for training and seven for testing, while on the CMU-PIE database a *one-shot training* scenario is evaluated on twenty test images for each person.

The upright descriptors performed best. The results confirm, that matching based on subregions improves the result over all-to-all matching. Outlier removal based on RANSAC further increases the performance even for matching on a regular grid. The combined system of upright SIFT and SURF descriptors achieved the best results.

### 2.3.3 Face Detection

Though many approaches to face detection using local features and to object detection using interest points exist, there has been to our knowledge only one approach to face detection involving interest points. As these ideas will form the basis of our approach, they are described in particular detail.

Toews and Arbel propose in [TA06, TA07, TA09] a probabilistic face appearance model which is - to some extent - invariant to changes in pose (and view-point). It basically consists of three componants:

1. A set of scale invariant features (e.g. SIFT)

2. An *object class invariant* (OCI)

3. A statistical relationship between the feature set and the OCI

**The set of features:** A feature $m_i : (m_i^a, m_i^g, m_i^b)$ is defined by an appearance $m_i^a$, a geometry $m_i^g$, and a binary component $m_i^b$. The appearance component specifies the appearance of a feature in terms of its feature descriptor (e.g. the 128-dimensional SIFT descriptor). The geometry $m_i^g : (x, y, \sigma_i, \theta_i)$ holds its

[4] Available at: http://www.ece.osu.edu/ aleix/ARdatabase.html

Figure 2.1: Example OCI: An OCI is represented as a vector from the base of the nose to the

2D position, scale and angle. The binary component $m_i^b$ indicates whether the corresponding feature is present or not.

**The object class invariant:** The authors define an OCI as

> an abstract 3D geometrical structure defined with respect to an underlying 3D object class, whose projection in the image plane maintains a consistent geometrical interpretation across different viewpoints and object class instances [TA09].

A 3D vector from the base of the nose to the forehead as illustrated in Figure 2.1 maintains this property as long as the camera does not move under or over the head. In these cases we could not uniquely determine the vertical angle between the camera and the face (and thus the real distance between the base of the nose and the forehead). Hence the choice of an OCI is dependent on the desired degrees of freedom. These structures need to be labeled by hand as it is difficult to derive them directly from images.

An OCI is denoted as $o : (o^g, o^b)$, having a geometry and a binary component. It has no appearance description, since it is not actually visible in the scene.

**The probabilistic relationship:** The relationship between an OCI and a set of features is modeled as the conditional probability of the OCI given the set of features. This way, the probability of an OCI hypothesis can be expressed in terms of an observed set of features. Following Bayes rule, the posterior probability of an OCI given a set of features is proportional to the probability of the feature set given the OCI and the prior probability of the OCI:

$$p(o|\{m_i\}) = \frac{p(\{m_i\}|o)\, p(o)}{p(\{m_i\})} = p(o)\frac{\prod_i p(m_i|o)}{p(\{m_i\})}, \tag{2.32}$$

assuming conditional feature independence to keep the model tractable. The term $p(o)$ is the prior probability of the OCI and $p(\{m_i\})$ is a normalization coefficient ensuring that $p(o|\{m_i\})$ is a valid probability distribution. The remaining term $p(m_i|o)$ models the probability of a specific feature given the OCI.

$$p(m_i|o) = p(m_i^a|m_i^b)\, p(m_i^b|o^b)\, p(m_i^g|o^b, o^g), \tag{2.33}$$

where $p(m_i^a|m_i^b)$ models the conditional probability of feature appearance given feature occurrence and $p(m_i^g|o^b, o^g)$ the conditional probability of feature geometry given OCI occurrence and OCI geometry. Those distributions are modelled as Gaussians. The probability of feature occurrence given OCI occurrence $p(m_i^b|o^b)$ is a binomial distribution.

**Model training:** In order to find model features which are, based on their appearance, distinctive for their relative position, scale and angle to the OCI, mean-shift clustering [FH75] is applied, starting with the whole set of training features as potential model features. Unfortunately, the authors provide no further information as to how they apply this procedure. The likelihood ratio $\frac{p(m_i^{b=1}|o^{b=1})}{p(m_i^{b=1}|o^{b=0})}$ is used to evaluate the distinctiveness of a model feature for its respective location. The cluster represented by a feature $m_i$ is delimited by a global geometrical threshold $T^g$ and a feature specific appearance threshold $T_i^a$ chosen as to maximize the feature's distinctiveness. After model training, features with low distinctiveness are removed from the model.

**Detection/Localization:** For detection, all features extracted from a test image are matched to all model features using the previously defined thresholds $T^g$ and $T_i^a$. Each single feature match *suggests* the geometry of an OCI, using

the model feature geometry which is defined relatively to the OCI. A cluster of such geometries further suggests the presence of a real OCI. An hypothesis is tested using Bayes' decision ratio

$$\gamma(o^g) = \frac{p(o^g, o^{b=1}|\{m_i\})}{p(o^g, o^{b=0}|\{m_i\})} = \frac{p(o^g, o^{b=1})}{p(o^g, o^{b=0})} \prod_i \frac{p(m_i|o^g, o^{b=1})}{p(m_i|o^g, o^{b=0})}, \tag{2.34}$$

determining the likelihood ratio of a true vs. false OCI hypothesis given the observed feature set. The first terme on the right describes the prior ratio of OCI occurrences and is manually set to regulate the ratio of true vs. false positive detector responses. The second term, $\prod_i \frac{p(m_i|o^g, o^{b=1})}{p(m_i|o^g, o^{b=0})}$ represents the likelihood ratio of valid to invalid feature matches and reveals an inconsistency of the model, when it is disassembled according to Equation (2.33):

$$\begin{aligned} \frac{p(m|o^g, o^{b=1})}{p(m|o^g, o^{b=0})} = & \quad \frac{p(m^a|m^{b=1})}{p(m^a|m^{b=1})} \cdot \frac{p(m^{b=1}|o^{b=1})}{p(m^{b=1}|o^{b=0})} \cdot \frac{p(m^g|o^{b=1}, o^g)}{p(m^g|o^{b=0}, o^g)} \\ = & \quad \frac{p(m^{b=1}|o^{b=1})}{p(m^{b=1}|o^{b=0})} \cdot \frac{p(m^g|o^{b=1}, o^g)}{p(m^g|o^{b=0}, o^g)}, \end{aligned} \tag{2.35}$$

where index $i$ has been removed for better readability. In this case, the likelihood ratio of appearances collapses to one. Thus we suggest a reformulation of Equation (2.33) as to define a conditional probability of feature appearance given OCI occurrence:

$$p(m_i|o) = p(m_i^a|o^b)\, p(m_i^b|o^b)\, p(m_i^g|o^b, o^g) \tag{2.36}$$

**Discussion:** The evaluation in [TA06] shows a maximum detection rate of 81%, which is aproximately achieved from a false detection rate of 10% onwards.

In the context of gender classification the autors use this model as a preprocessing step to filter face specific features from a cluttered scene image [TA09]. Using solely features which are good for representing a general face may not lead to satisfactory results in the context of face identification as distinctive features with respect to a particular person may be already discarded in this first stage. We specifically adress this question in our model, which is presented in the following chapter.

# 3 A Scale Invariant Probabilistic Model For Face Recognition

Its applicability in realistic scenarios is the main design goal for our face recognition model. We outline our understanding of a *realistic scenario* by the properties, which we assume to have the most significant impacts on design decisions: A face can be present anywhere in the scene. Furthermore, a face can be of any size and appear in any relative 3D-position to the camera. Secondly, there may be complex backgrounds and thus the model needs to be robust against clutter.

As seen in the previous section, parts based object models have the great advantage of being robust against partial occlusion (e.g. facial hair, glasses or other objects). We choose to use *SIFT features* to model these parts, because they are scale invariant by definition, and beyond that, they proved to be rather insensitive to small changes in pose and illumination [MS05].

The superiority of *probabilistic models* over deterministic models in the context of object detection has been shown by Schmid in [Sch99]. The promising results on face detection presented by Towes and Arbel [TA06, TA07, TA09] further suggest to opt for a probabilistic face detection model. An inherent advantage of probabilistic models is the simplicity with which a background model can be incorporated.

### Multi View vs. View Invariant Model

There are two complementary ways to achieve the capability of detecting objects across different poses and camera positions. The multi view approach follows the idea that the detection problem can be subdivided into detecting different views separately. In practice, multiple detectors - one for each *distinct* view - are trained and applied to sample images more or less separately [Low01]. Therefore, the learning and detection procedures are rather complex and time consuming. Finding distinct views from sample images requires large amounts

of training data. Furthermore, it is rather difficult for systems following this approach to deal with faces whose pose lies somewhere in between these distinct views, as multiple detectors close to this pose are expected to respond with little confidence.

A view invariant approach on the other side, models the appearance of an object class such as faces independently of its pose. Rather than explicitly accounting for pose variations on the top level, a single representation using perspective invariants is defined, which implicitly incorporates appearance variations due to different object poses [TA06].

### Geometrical Interdependencies in Models Based on Local Features

Modeling faces from local features imposes the question of how to represent the geometrical relationship of features to each other and how to localize a face based on feature locations. A *bag of features* approach does not include any geometrical interdependencies. Thus, the 2D location of an instance of the object class modeled can only be inferred by identifying modes of the model feature density in a test image. As there is no knowledge about relative feature scales and angles, it is not possible to determine these properties for the underlying object class either. This approach requires a rather large number of model features (or features with a high conditional occurrence probability) but is very flexible with respect to object geometry.

If we interpret features as nodes in a graph, where arcs denote a modeled geometrical dependency, the degree of connectivity lets us characterize all methods, which incorporate feature geometry to some extent. A *fully connected graph* represents the most complex model in this regard, as it has an extremely high number of parameters. For probabilistic models consisting of thousands of parts, model learning with this approach can be considered infeasible. Additionally, such a model imposes very restrictive priors to the feature geometry given the object. Therefore it is only applicable to very rigid objects.

As proposed by Carneiro and Lowe [CL06], *k-connected geometric models* are more flexible. With this approach, a feature geometry depends only on the k closest features, which drastically decreases the model complexity (obviously depending on k). These models are globally more flexible while feature geometries are locally bound by features in their neighborhood.

The most simple way of modeling geometrical relationships in parts based

models has a star shaped structure. In this case, the relationship between all features and an (arbitrary) *super feature* are learned, which further increases the potential geometrical flexibility and decreases the number of model parameters to be learned. The right choice of the reference frame (i.e. the central feature) is crucial.

In the context of face recognition, some degree of local geometrical flexibility is desirable, as relative feature positions vary with a change in facial expression as well as between faces of different individuals. The method proposed by Towes and Arbel using the *Object Class Invariant* (OCI) combines both desirable characteristics: The face model is reported to be invariant to camera viewpoint to some extent, and feature geometries are solely defined with respect to the OCI, representing a star shaped structure. Therefore, we base our model on this approach.

## 3.1 Detection Model

Similar to [TA06], we describe the appearance of a face in terms of a set of model fetures $\{f_i\}^{model}$ and an OCI $o$. Additionally, we distinguish between two object classes $C := \{face, bg\}$, representing faces and the background, and we define a probabilistic relationship between model features, the OCI and the object classes.

### Object Class Invariant

Although the general idea of an OCI has already been introduced in Section 2.3.3, at this point the concept should be explained in some more detail. The purpose of the OCI is to serve as a scale invariant reference frame for the position of a face, in order to associate features with relative locations to this reference frame. This conceptual reference frame is defined in 3D world coordinates as it refers to 3D object classes (e.g. faces) and not directly to their 2D projections. To be able to extract the same amount of information, that is the pose of the underlying object class, from the 2D projection of the reference frame in an images as from its three dimensional counterpart, it has to be ensured that this reference frame is not subject to any perspective distortion.

Consider an OCI defined by three orthogonal vectors in 3D space and its two dimensional projection from two different viewpoints. We can not uniquely

determine the correct corresponding vectors in these two images, and the OCI geometry is distorted in a way that a unique geometrical interpretation is impossible. Thus we might choose a simpler OCI geometry.

Towes and Arbel propose to use a vector from the base of the nose to the forehead [TA06] as a face specific OCI. Recall that the definition of the OCI is in 3D world coordinates. The projection of this vector onto the image plane, however, permits a consistent interpretation with its definition in three dimensional space[1] as long as the camera does not move over or under the head, in which case the OCI would be subject to perspective distortion. This kind of OCI offers several degrees of freedom: Faces may be located at varying positions, of different sizes and of different in-plane orientations, as the position, the length and the orientation of the vector defining the OCI allow to marginalize out these variations. Additionally, rotating the face about the vertical axis does not affect the geometry (and thus the interpretation consistency) of the OCI. As with this kind of rotation, facial feature locations relative to the OCI vary in horizontal direction, the challenge for a view invariant face detection system is to find facial features which are distinctive for their relative geometry across views and subjects.

In our model, we describe an OCI by its geometry $g_o : (x_o, y_o, \sigma_o, \theta_o)$, defining its position in the image, its scale and angle.

### Model Feature Representation

A model feature $f_i : (a_i, T_i^a, g_i, p_i^{face}, p_i^{bg})$ comprises the following components:

1. The feature appearance $a_i$ is defined by its 128 dimensional SIFT descriptor.

2. As a model feature represents a cluster of similar SIFT features, a distance threshold $T_i^a$ delimits the *basin of attraction* of the model feature $f_i$ in appearance space.

3. The feature geometry $g_i : (x_i, y_i, \sigma_i, \theta_i)$ describes its respective 2D position, scale and angle, which we also get from the SIFT feature extraction process (Section 2.3.1). Again, as we deal with feature clusters, the range of geometries is delimited by a set of scale invariant thresholds

[1] assuming orthogonal projection

$T^g : (T^{pos}, T^\sigma, T^\theta)$. Rather than being feature specific, these thresholds are defined globally.

4. The entity $p_i^{face}$ describes the fraction of face training features, which correspond to this particular model feature.

5. Similarly, $p_i^{bg}$ is the fraction of background training features corresponding to the model feature $f_i$.

### Probabilistic Relationship

In order to formulate a probabilistic relationship between object classes, model features and the OCI, we adapt the standard formulation of a generative object detection model (introduced in Section 2.3.1) to our notation. Given a set of features$\{f_i\}, i = 1, \cdots, N$ and an object class $c \in C$, their probabilistic relationship is defined as

$$P(c|\{f_i\}) = \frac{P(\{f_i\}|c)\, P(c)}{P(\{f_i\})} \approx \prod_{i=1}^{N} P(f_i|c), \tag{3.1}$$

using Bayes theorem and assuming conditional feature independence (compare Equations (2.19) and (2.20)). Thus, we concentrate on the individual feature probability conditioned on both the face ($face$) and the background ($bg$) class. The characteristics we want to incorporate are the feature appearance $a_i$ given a class $c$, the occurrence probabilities $p_i^{face}$ and $p_i^{bg}$, as well as the feature geometry $f_g$. With respect to the OCI, it becomes clear that the individual feature probabilities for the classes $face$ and $bg$ are different: While for the $face$ class we aim to model feature geometries relatively to the OCI $g_o$, features in the background can be assumed to be independent of the presence and thus the geometry of an OCI. Therefore, we derive a specific formulation for the individual feature probabilities for each object class:

**Face Feature Probability:** Given a model feature $f_i : (a_i, T_i^a, g_i, p^{face}, p^{bg})$ and an OCI $g_o$, the individual face feature probability is defined as

$$p(f_i|face) = p(a_i|face) \cdot p(g_i|face, g_o) \cdot p_i^{face} \tag{3.2}$$

The term $p(a_i|face)$ describes the probability of model feature $f_i$ having the

appearance $a_i$ if it represented a $face$ feature. We assume a Gaussian distribution in appearance space. The distribution of feature geometry relatively to the OCI $p(g_i|face, g_o)$ is assumed to be Gaussian as well. In Section 3.1.1 we describe how to represent the relationship between feature geometry and OCI geometry. Section 3.1.2 discusses how the parameters for the appearance and geometry distributions are derived from data.

**Background Feature Probability:** As mentioned above, in the case where a feature $f_i$ corresponds to the background $bg$, there is no need to model its geometrical relationship to the OCI. Hence, the formula reduces to

$$p(f_i|bg) \;=\; p(a_i|bg) \;\cdot\; p(g_i|bg) \;\cdot\; p_i^{bg} \tag{3.3}$$

The appearance probability $p(a_i|bg)$ is in this case also modeled as a Gaussian distribution. As a background feature can potentially appear anywhere in the image, the geometrical feature probability for the class $bg$ is modeled as a uniform distribution.

### 3.1.1 Geometrical Relationships

Before describing the learning and detection procedures, it is useful to take a look at the mathematical background of the geometrical relationships involved. As mentioned above, our geometrical model is based on a star shaped structure, where all feature geometries are modeled relative to the OCI. Hence, given a feature geometry $g_i : (x_i, y_i, \sigma_i, \theta_i)$ and an OCI geometry $g_o : (x_o, y_o, \sigma_o, \theta_o)$ in absolute image dimensions, we need to normalize $g_i$ to $g_o$, that is to determine a relationship $g_i^r : (x_i^r, y_i^r, \sigma_i^r, \theta_i^r)$, which describes the feature geometry in function of a normalized OCI geometry $g_o^{norm} : (x_o = 0, y_o = 0, \sigma_o = 1, \theta_o = 0)$. In practice, it turns out to be more useful to normalize the OCI geometry to feature geometry. This way we are able to easily predict an OCI based on the geometry of a feature in a test image which corresponds to a model feature. Therefore, we estimate the relationship $g_i^r : (x_i^r, y_i^r, \sigma_i^r, \theta_i^r)$ based on a normalized feature geometry $g_i^{norm} : (x_i = 0, y_i = 0, \sigma_i = 1, \theta_i = 0)$.

**Normalization to Feature Geometry:** This normalization is performed in three steps. We start by defining the relative position:

$$x_i^{shift} = x_o - x_i \qquad\qquad y_i^{shift} = y_o - y_i \tag{3.4}$$

With $x_i^{shift}$ and $y_i^{shift}$ , we describe the relationship of $g_o$ to a feature geometry $g_i : (x_i = 0, y_i = 0, \sigma_i, \theta_i)$, which allows us to predict the 2D location of the OCI based on $g_i^{shift}$ and the image location of a test feature, which matches model feature $f_i$. The relationship $g_i^{shift}$ can thus be seen as a shift invariant mapping from the model feature $f_i$ to the OCI. We still need to obtain scale and rotation invariance. The order of these two normalization operations does not affect the result. We start with scale invariance:

$$\sigma_i^{scale} = \frac{\sigma_o}{\sigma_i} \qquad x_i^{scale} = \frac{x_i^{shift}}{\sigma_i} \qquad y_i^{scale} = \frac{y_i^{shift}}{\sigma_i} \tag{3.5}$$

The geometry $g_i^{scale}$ represents a mapping from a feature geometry $g_i : (x_i = 0, y_i = 0, \sigma_i = 0, \theta_i)$ to the OCI. Rotation normalization involves projecting relative OCI coordinates to local feature coordinates:

$$\begin{aligned} \sigma_i^{rot} &= \sigma_i^{shift} && (3.6)\\ \theta_i^{rot} &= \theta_o - \theta_f \\ x_i^{rot} &= cos(-\theta_i) \cdot x_i^{scale} - sin(-\theta_i) \cdot y_i^{scale} \\ y_i^{rot} &= sin(-\theta_i) \cdot x_i^{scale} + sin(-\theta_i) \cdot y_i^{scale} && (3.7) \end{aligned}$$

Based on this final mapping $g_i^r = g^{rot}$, we are able to define powerful transformations and expressions: It is possible to project feature coordinates into (normalized) OCI coordinates, which will be useful to estimate a feature bounding box. We can compare the relative position of a pair of features to each other through comparing their respective mappings to the OCI. And based on the absolute geometry of a feature in a test image, we can project an OCI into the image.

**Geometrical Agreement:** In order to determine model features which are distinctive for their relative position to the OCI, we need a function to quantify the

relative position of a pair of features $(f_1, f_2)$ to each other, which we call *geometrical agreement* [TA06]. Given a normalized feature geometry $g_i^r$ corresponding to feature $f_i$ and a set of geometrical thresholds $T^g : (T^{pos}, T^{scale}, T^{angle})$, then a feature $f_j$ with geometry $g_j^r$ *agrees geometrically* with $f_i$, if it lies within the four dimensional scale invariant bounding box defined by $T^g$ around feature $f_i$. More formally, $f_j$ agrees geometrically with $f_i$ if and only if the following expression is true:

$$\begin{aligned} GeoAgg(f_i, f_j) = &( \mid x_i - x_j \mid < T^{pos} \cdot \sigma_i ) \wedge \\ &( \mid y_i - y_j \mid < T^{pos} \cdot \sigma_i ) \wedge \\ &( \mid log\, \sigma_i - log\, \sigma_j \mid < log\, T^{scale} ) \wedge \\ &( \mid \theta_i - \theta_j \mid < T^{angle} ) \end{aligned} \tag{3.8}$$

Note that position is compared relatively to feature normalized OCI scale $\sigma_i$ and scale difference is evaluated in the log domain, as differences in this dimension grow exponentially with absolute scale. By definition, geometrical agreement is a binary function. We will show that despite its simplicity, this function is sufficient for our model.

**OCI Prediction Based on a Single Feature** Consider a test feature $f_j$ in an image which corresponds to a face model feature $f_i$. If the normalized relationship $g_i$ between the model feature $f_i$ and the OCI geometry is known, the OCI can be projected into the image plane of the test image based this relationship $g_i$ and the test feature geometry $g_j$. Recall that $g_i$ defines a mapping from the normalized model feature geometry $g_i^{norm} : (x_i = 0, y_i = 0, \sigma_i = 1, \theta_i = 0)$ to the OCI. Applying the inverse functions of the shift, scale and orientation normalizations as defined in Equations (3.4), (3.5) and (3.7), respectively, in inverse order, we can predict the OCI geometry $g_o$ in absolute image coordinates as follows: Starting by rotating $g_i$ into image coordinates, we get

$$\begin{aligned} x_o^{inv\_rot} &= cos(\theta_j) \cdot x_i - sin(\theta_j) \cdot y_i \\ y_o^{inv\_rot} &= sin(\theta_j) \cdot x_i + cos(\theta_j) \cdot y_i \\ \theta_o^{inv\_rot} &= \theta_j + \theta_i \end{aligned} \tag{3.9}$$

Now, the OCI geometry $g_o^{inv\_rot}$ needs to be scaled and shifted approriately:

$$x_o^{inv\_scale} = x_o^{inv\_rot} \cdot \sigma_j \qquad y_o^{inv\_scale} = y_o^{inv\_rot} \cdot \sigma_j \qquad \sigma_o^{inv\_scale} = \sigma_i \cdot \sigma_j \quad (3.10)$$

$$x_o^{inv\_shift} = x_o^{inv\_scale} + x_j \quad y_o^{inv\_shift} = y_o^{inv\_scale} + y_j \quad (3.11)$$

The final OCI geometry $g_o$ in absolute image coordinates is given by $g_o : (x_o^{inv\_shift}, y_o^{inv\_shift}, \sigma_o^{inv\_scale}, \theta_o^{inv\_rot})$. This procedure will be used to construct face hypotheses in Section 3.1.3.

**Predicting the Feature Location** In the context of detection and identification system combination in Section 3.3, we are interested the exact opposite normalized geometrical relationship between the OCI and model features, namely the position $g_i$ relatively to a normalized OCI $g_o^{norm} : (x_o = 0, y_o = 0, \sigma_o = 1, \theta_o = 0)$. Here we will show how we can easily transform the first normalized form into the other. The initial relationship $g_i$ defines the relationship between $g_i^{norm} : (x_i = 0, y_i = 0, \sigma_i = 1, \theta_i = 0)$ and any relative OCI geometry $g_o : (x_o, y_o, \sigma_o, \theta_o)$. The idea behind this transformation is to invert all relative coordinates (i.e. position, scale and angle). As the inversion of relative orientation includes a rotation of x- and y-coordinates, we accomplish the whole transformation in two steps:

$$x_i' = \frac{-x_i}{\sigma_i} \qquad y_i' = \frac{-y_i}{\sigma_i} \qquad \sigma_i' = \frac{1}{\sigma_i} \quad (3.12)$$

$$\begin{aligned} x_i'' &= cos(-\theta) \cdot x_i' - sin(-\theta) \cdot y_i' \\ y_i'' &= sin(-\theta) \cdot x_i' + sin(-\theta) \cdot y_i' \\ \theta_i'' &= -\theta_i \end{aligned} \quad (3.13)$$

the final feature geometry normalized to the OCI is given by $g_i^{-1} : (x_i^{-1} = x_i'', y_i^{-1} = y_i'', \sigma_i^{-a} = \sigma_i', \theta_i^{-1} = \theta_i'')$. As the transformations introduced above form the basic concepts of our scale invariant face recognition model. Therefore, it is important to introduce them in such detail.

### 3.1.2 Model training

Model training aims to identify clusters of SIFT features which are not only distinctive for faces but also for their respective location relative to the OCI. For these clusters, we further need to estimate their appearance and geometry distributions as well as to determine the feature specific appearance threshold $T_i^a$ and the binomial feature occurrence probabilities $p_i^{face}$ and $p_i^{bg}$.

#### Feature Clustering

Towes and Arbel propose in [TA09] for their probabilistic face detection model using the OCI a mean shift clustering approach [FH75] to determine distinctive model features.

They start with the whole set of face training features as potential clusters and proceed as follows: For each potential cluster, the appearance distance and geometrical agreement with all other training features are calculated. Based on these estimations, a feature specific appearance threshold is chosen as to maximize the probability ratio of geometrically agreeing vs. geometrically disagreeing face features, which the authors call feature distinctiveness. Then, the cluster center in appearance space and image space is set to the mean appearance and geometry of all geometrically agreeing features, which also *agree* in terms of appearance, that is whose distance in appearance space from the current cluster mean is smaller than the chosen appearance threshold. This procedure is carried out on all potential clusters and iterates until convergence (or until a maximum number of iterations is reached). However, the authors state that a single iteration may be sufficient.

After evaluating all potential clusters, an *independent subset* of clusters is determined and all other clusters as well as clusters with low distinctiveness are discarded. The authors give no further information on how this independent subset is determined. We suppose that, given any two clusters which agree geometrically and overlap in terms of appearance based on their appearance thresholds, the feature which is more *distinct* is kept and the other one discarded.

Besides this ambiguity, the proposed approach has from our point of view some other shortcomings: A mean shift procedure on both appearance and geometry is not guaranteed to converge, as geometrical thresholds are used to select features which potentially support a cluster. By *supporting features*

we mean features which agree in terms of appearance and geometry with a cluster. A feature, which has been a positive (the cluster supporting) sample in the last iteration, may now disagree geometrically and thus may represent a negative sample, which is now tried to be excluded, based on the way appearance thresholds are determined. Furthermore, since clustering starts with a large number of potential model features and clusters are only removed after all of them have been evaluated, this method seems to be rather time consuming and inefficient.

Based on these reflections we opt for a different approach: At first, we construct clusters solely based on appearance. In a second step, these clusters are either selected to form a model feature or discarded if most of the supporting training features disagree in terms of their geometry relative to the OCI.

**Appearance Clustering:** Training features are clustered based on their appearance using an agglomerative clustering approach. Therefore, we start with all face training features as potential appearance clusters. We define a global appearance distance threshold $T^d$. Distances between SIFT descriptors are calculated using the Euclidean distance. In each iteration, we merge the pair of clusters with minimum Euclidean distance until there is no pair of clusters left, whose distance is lower than the predefined threshold $T^d$. For each cluster, we keep a counter of supporting training features. A merge is performed by estimating the mean SIFT feature vector of both clusters, where the contribution of each cluster is weighted by the number of its respective supporting training features. Both original clusters are considered as being *invalid* for future merging operations, while the supporting features of both clusters are associated with the derived cluster and its feature counter is set appropriately.

In order to find the pair of clusters with minimum distance efficiently, we use a priority queue. Each queue entry represents a pair of clusters. The distance between their respective SIFT descriptors forms the key by which entries are prioritized. In order to initialize the queue, distances between all pairs of initial clusters are estimated. If the distance between a pair of clusters does not exceed the threshold $T^d$, a queue entry is created consisting of the estimated distance and the indices of the corresponding clusters. Note that with this method, there may be entries in the queue which are invalid. This happens when one of the corresponding clusters has been merged with another cluster after the

queue entry has been created. Therefore, finding the current pair of clusters with minimum appearance distance involves not only popping the top entry off the queue but also checking whether both corresponding clusters are still valid. Otherwise the entry is discarded.

To be able to avoid reaching memory limitations, we developed a multi-tier appearance clustering approach: Instead of using a single appearance threshold $T^d$, we define a progression of thresholds $(T_1^d, \cdots, T_N^d)$. The algorithm outlined above starts with the first threshold in this sequence $T_1^d$. Once it terminates (i.e. there are no more pairs of valid clusters whose distance is lower than $T_1^d$), the algorithm is re-initialized with the remaining valid clusters and the next appearance threshold in the sequence. This process iterates for each threshold $T_i^d, i = 1 \cdots, N$.

We choose to set the final appearance threshold $T_N^d$ as to maximize the number of appearance clusters with at least two supporting training features (on the data sets we used for empirical evaluation, $T_{max}^d \approx 0.5$ was independent of the number and choice of training features).

**Geometry Clustering:** After appearance clustering, all clusters with less than two support features are discarded. The remaining clusters are inspected individually. Given an appearance cluster $c_i$ with supporting features $f_i^j$, we seek to determine whether for any geometry $g_i^j$, more than half of the support features *agree geometrically*. Otherwise, the appearance cluster $c_i$ is considered to be not distinctive for any geometry and is discarded. Thus, we count for each geometry $g_i^j$ the number of geometrically agreeing features $f_i^k, k \neq j$. If the condition defined above holds for any geometry $g_i^j$, this geometry serves as a preliminary cluster geometry $g_i$. The set of remaining clusters forms the set of model features $\{f_i\}$.

### Estimation of Model Parameters

For each of the derived model features $f_i$ we need to determine the following parameters:

- The feature specific appearance threshold $T_i^a$
- The appearance distribution $p(a_i|c), c \in C$
- The geometrical distribution relative to the OCI $p(g_i|face, g_o)$

- The binomial feature occurrence probabilities $p^{face}$ and $p^{bg}$

We define the set of supporting features of a model feature $f_i$ as the set of features $\{f_i^j\}$ which have been merged in the course of building this particular model feature and which agree with its preliminary geometry.

**Appearance Threshold:** The only parameter associated with a model feature $f_i$ which is independent of any object class is the feature specific appearance threshold $T_i^a$. However, as our goal is to create model features which are distinctive for faces and their respective location, we incorporate only support features $\{f_i^j\}$ for estimating this parameter. The threshold $T_i^a$ is set to the maximum distance of all support features to the model feature appearance

$$T_i^a \; = \; argmax_j \, d(a_i^j, a_i), \tag{3.14}$$

where $d(a, b)$ denotes the Euclidean distance between $a$ and $b$. This way we choose the smallest possible appearance threshold to include all support features and thus maximize the probability of excluding features corresponding to the background or face features which disagree with the model feature geometry while preserving the features descriptiveness.

**Geometry Distribution:** As mentioned above, for the relative feature geometry to the OCI we assume a Gaussian distribution. Thus, we compute the mean and diagonal covariance of the geometries $\{g_i^j\}$ corresponding to the set of support features $\{f_i^j\}$.

The uniform geometrical distribution for any feature given the background class $bg$ is dependent of the size of the image in pixels $A_I$ and the range of scales in which SIFT descriptors have been extracted $\sigma_I$:

$$p(g_i|bg) = \frac{1}{A} \cdot \frac{1}{\sigma} \cdot \frac{1}{2\pi} \tag{3.15}$$

Rather than being determined in the learning stage, this uniform distribution is dependent on the test setting (e.g. the test image resolution). As we can see, this distribution is equal for any feature $f_i$.

**Appearance Distributions:** Based on the way we designed the clustering algorithm, in this parameter estimation stage there are at least two face support

features for each model feature $f_i$. Actually, a *large majority* of the derived model features happens to be supported by a very small number of training features. Modeling a Gaussian appearance distribution with a diagonal covariance of 128 dimensions (the dimensionality of SIFT feature vectors) using such a small number of samples yields a very inaccurately estimated parameter which would result in an over fitted model to the training data. Therefore, we estimate a single distance variance parameter between support feature appearance $a_i^j$ and mean model feature appearance $a_i$.

The estimation of the appearance distribution given the background proceeds in a similar way. At first, we determine which features of a set of background training features $\{f_k\}^{bg}$ agree with the model feature $f_i$ in terms of appearance, that is $d(f_k^{bg}, f_i) \leq T_i^a$. Then we compute the distance variance corresponding to the background using this subset of background training features.

**Binomial Feature Probability:** The class dependent feature occurrence probability $p_i^{face}$ is determined as the number of supporting face features over the total number of face training features. Similarly, the background occurrence probability $p_i^{bg}$ is set to the fraction of background features which agrees with $f_i$ in terms of its appearance. In addition to that, we consider all face features not agreeing geometrically with $f_i$ as a background feature.

Comparative evaluation of this approach with the one proposed by Towes and Arbel [TA09] in an early design stage showed that our two step agglomerative clustering approach yields similar performance results using far less model features. Furthermore, our method is straight forward to implement and easily extendable to multiple object classes.

### 3.1.3 Face Detection and Localization

Face detection utilizes a set of SIFT descriptors $\{f_j\}^{test}$ extracted from a test image and the trained model in order to construct hypotheses of OCI geometries. Finally, these hypotheses are validated, which is the decision of whether the hypothesis is more likely to correspond to an actual face or the background.

### Hypothesis Construction

Given a set of model features $\{f_i\}^{model}$ and a set of test features $\{f_j\}^{test}$, we start by trying to match all test features with all model features. A test feature $f_j^{test}$ is said to successfully *match* a model feature $f_i^{model}$ if its distance to the mean appearance $a_i^{model}$ does not exceed the feature specific appearance threshold $T_i^a$.

$$match(f_j^{test}, f_i^{model}) = d(a_j^{test}, a_i^{model}) < T_i^a \tag{3.16}$$

For each successful match between a test feature $f_j^{test}$ and a model feature $f_i^{model}$, we *predict* the geometry of an OCI $g_o^{ij}$ based on the test feature geometry $g_j^{test}$ in absolute image dimensions and the normalized OCI geometry $g_i^{model}$ as shown in Section 3.1.1. Based on the set $\{g_o^{ij}\}$ of predicted OCI geometries, we aim to generate a set of hypotheses $\{H_k\}$. Therefore, we determine for each predicted OCI geometry $g_o^{ij}$ the subset of all predictions $\{g_o^{ij}\}$, which agrees geometrically with $g_o^{ij}$. The sets of test features $\{f_j\}^{test}$ and model features $\{f_i\}^{model}$, which correspond to geometrically agreeing OCI predictions, together with the currently evaluated OCI geometry $g_o^{ij}$ form an hypothesis $H_k$. Thus, we generate one hypothesis for each successfully matched test feature $f_j$ by associating all other successfully matched test features and their corresponding model features with the predicted OCI $g_o^{ij}$, whose OCI predictions agree with this geometry.

### Hypothesis Validation

An hypothesis $H : (\{f_j\}^{test}, \{f_i\}^{model}, g_o)$, which associates a set of features $\{f_j\}^{test}, j = 1, \cdots, M$ with a set of face model features $\{f_i\}^{model}, i = 1, \cdots, N$ and predicts an OCI geometry $g_o$, can be validated using the likelihood ratio of true vs. false feature correspondences:

$$\begin{aligned}
\gamma(H) &= \frac{p(face|\{f_i\})}{p(bg|\{f_i\})} \\
&= \frac{p(\{f_i\}|face)\, p(face)}{p(\{f_i\}|bg)\, p(bg)} \\
&= \frac{p(face)}{p(bg)} \cdot \prod_{i=1}^{N} \frac{p(f_i|face)}{p(f_i|bg)},
\end{aligned} \tag{3.17}$$

where we used Equation (3.1). An hypothesis is accepted if $\gamma(\{f_i\})$ is greater (or equals to) one. This equation shows that the approximation of $p(c|f_i)$ in Equation (3.1) turns into an equality, when used in the context of likelihood ratios, by adding the term $\frac{p(face)}{p(bg)}$, which theoretically describes the likelihood ratio of the class prior probabilities. Practically, we do not know in advance, features of which class are more likely to appear. Hence, there are two possibilities to model these priors: Either we assume that both classes are equally likely, in which case the term $\frac{p(face)}{p(bg)}$ disappears from the formula; or we manually set this prior ratio to an arbitrary value, which allows us to empirically determine a good compromise between false positive and false negative detector responses (see Section 4.1). Using the individual feature probabilities for faces (Equation (3.2)) and background features (Equation (3.3)), the decision ratio defined above expands to

$$\gamma(H) = \frac{p(face)}{p(bg)} \cdot \prod_{i=1}^{N} \frac{p(a_i|face)}{p(a_i|bg)} \cdot \frac{p(g_i|face, g_o)}{p(g_i|bg)} \cdot \frac{p_i^{face}}{p_i^{bg}}, \tag{3.18}$$

which consists of four distinct terms: the class prior ratio discussed above, and for each individual feature, terms describing the class likelihood ratio based on either feature appearance or feature geometry or feature occurrence probabilities.

In practice, we do not *accept* all hypotheses $H_k$ generated as shown in the previous section, which would theoretically pass the validation. That is because a single test feature $f_j^{test}$ may support more than one hypothesis. Either by matching more than one model feature, or by predicting an OCI geometry, which agrees geometrically with more than a single hypothesized OCI. However, we start by evaluating the probability ratio $\gamma(H_k)$ for each hypothesis. The hypothesis $H_k^{max}$, which passes the validation and has the highest likelihood ratio of being a true face vs. corresponding to background clutter, is accepted and removed. Then, all test features $f_j^{test}$, which are associated with $H_k^{max}$, as well as their predicted OCI geometries $g_o^{ij}$, are discarded. This involves updating the sets of associated test and model features for all remaining hypotheses $\{H_k\}$ and re-evaluating $\gamma(H_k)$. This procedure iterates until there is no more hypothesis, which passes validation.

## 3.2 Identification Model

As we focus on one shot learning, that is using only a singe training image per subject, it is only reasonable to define a *deterministic* identification model. Therefore, a basic model does not involve any model training. Instead, we simply store all SIFT features extracted from a training image $\{f_j\}^{training}$ as a person specific template $\{f_j\}^{temp}$ in a database. This database can thus be mathematically described as a set of templates $DB : \{\{f_j\}_i^{temp}\}$, for subjects $i = 1, \cdots, N$. Hence, we concentrate on defining a *matching strategy* on feature level, as well as defining *similarity measures* on subject level, that is a function $S(\{f_k\}^{test}, \{f_j\}^{temp})_i$, which describes the degree of similarity between test features and template features of subject $i$.

Identification involves evaluating the similarity between test features and all templates. The subject corresponding to the template with the highest similarity measure

$$argmax_i \left( S(\{f_k\}^{test}, \{f_j\}^{temp})_i \right), \tag{3.19}$$

is considered to be present in the image. An authentication system, which focusses on comparing a set of test features with a particular template $i$, incorporates a threshold $t_{auth}$ to take a binary decision of whether the subject corresponding to template $i$ is present in the image or not:

$$S(\{f_k\}^{test}, \{f_j\}^{temp})_i > t_{auth} \tag{3.20}$$

### 3.2.1 Potential Matching Feature Selection

The methods proposed by Bicego et al. (SIFT_GRID) in [BLGT06] and Luo et al. (SIFT_CLUSTER) in [LMT+07] showed, that a region based feature matching strategy has two advantages over matching all possible pairs of features:

Matching solely features of corresponding subregions in the image effectively reduces the number of false correspondences and hence increases the identification performance. As a positive side effect, this approach demands a significantly lower computation time on average, although the asymptotic time complexity stays unchanged ($O(NM)$, for $M$ features in the test image and $N$ features in the database).

A disadvantage of both methods mentioned above is, that they require test images to be aligned (i.e. to have a fixed position of and distance between the eyes) and rasterized (i.e. cropped to a predefined width and hight). These requirements are not met in a realistic scenario. As we finally aim to create a combined face detection and identification system, we use the geometry of the *Object Class Invariant* (OCI) to perform some kind of alignment. The OCI geometry holds rich information about the face location, pose and size, which we can easily exploit to adapt a region based matching approach:

Given a set of training features $\{f_j\}^{training}$ and the corresponding OCI $g_o$, we determine the normalized relationship between all training feature geometries $g_j$ and the OCI geometry $g_o$ as described in Section 3.1.1. Instead of storing the feature geometries in absolute image coordinates together with their respective features in the database, we replace them by their normalized counterparts. We similarly proceed with a set of test features: we replace the geometries defined in absolute image coordinates by their normalized geometrical relationship to the OCI $g_o^{test}$ in the test image. Due to the fact that all feature geometries - those derived from the test image as well as those stored in the database - are described with respect to a common reference frame, namely the OCI, we can now define a region based feature selection method:

Each test feature $f_k^{test}$ is matched with the subset of features of a template $\{f_j\}^{temp}$, whose normalized geometries $g_j^{temp}$ *agree* with the normalized geometry $g_k^{test}$ of the test feature. This region based matching strategy differs in several ways from the ones proposed by other authors: Rather than fixing the number of regions (i.e. the number of grid cells or the number of geometrical clusters), our approach keeps the *size* of a region fixed while the number of distinct regions is undefined. Furthermore, regions derived from geometrical agreement are not only bounded in 2D pixel coordinates but also in angle and scale dimensions. These additional restrictions further reduce the number of false correspondences. Assuming that the OCI geometry is known (either labeled by hand or located by our face detector), this region based matching strategy is applicable in uncontrolled environments without further alignment or registration.

### 3.2.2 Similarity Measures

How can we - based on a set of SIFT features representing the content of a test image and a set of test features representing the face of an individual - measure the similarity between this face model and the test image? A large variety of so called *similarity measures* have been proposed and experimented with in the literature (e.g. [BTG06, LMT+07, CSM08, DSC09, ]), some of which have been introduced in Section 2.3.2. As they have all been applied in different contexts, using different interest point descriptors or matching strategies, we intend to systematically explore the effects of various similarity measures on identification performance.

#### Voting Schemes

The simplest similarity measure we define, follows a voting scheme [Low04]. All test features $f_k^{test}$ are matched with all features $f_j^i$ of all templates $i$ in the database. For each test feature $f_k^{test}$, we determine the template feature $f_k^{min}$ with minimum Euclidean distance $d(f_k^{test}, f_j^i)$. We define a function $temp(f_k^{min})$, which maps the feature $f_k^{min}$ to the template $i$ that includes $f_k^{min}$. Additionally, we determine the template feature $f_k^{2nd}$, which has the second minimum distance to test feature $f_k^{test}$, and corresponds to any other template than $temp(f_k^{min})$. Then, we can define a function $vote(f_k^{test}, i)$, which is one if the template includes the feature $f_k^{min}$ with minimum distance to $f_k^{test}$ and the distance ratio $\frac{d(f_k, f_k^{min})}{d(f_k, f_k^{2nd})}$ does not exceed a threshold $T_r$

$$vote(f_k^{test}, i) = 1\left[\left(temp(f_k^{min}) = i\right) \wedge \left(\frac{d(f_k, f_k^{min})}{d(f_k, f_k^{2nd})} \leq T_r\right)\right], \tag{3.21}$$

where $1[expression]$ is one if the expression is true and zero otherwise. The distance ratio $\frac{d(f_k, f_k^{min})}{d(f_k, f_k^{2nd})}$ can be interpreted as a measure of ambiguity of a feature match. Thus, we ignore all test features $f_k^{test}$, which can not *uniquely* vote for a particular subject.

We may now define the similarity measure $S(\{f_k\}^{test}, \{f_j\}^{temp})_i^{VOTING}$ of a template $i$ to the whole set of test features $\{f_k\}^{test}$ as the number of times any

test feature *votes* for the template $i$:

$$S(\{f_k\}^{test}, \{f_j\}^{temp})_i^{\text{VOTING}} = \sum_k vote(f_k^{test}, i) \tag{3.22}$$

To account for different numbers of template features $f_j^i$ for each template $i$, we further define a *normalized* similarity:

$$S(\{f_k\}^{test}, \{f_j\}^{temp})_i^{\text{NORMALIZED}} = \frac{S(\{f_k\}^{test}, \{f_j\}^{temp})_i^{\text{VOTING}}}{|\{f_j^i\}|}, \tag{3.23}$$

where $|\{f_j^i\}|$ denotes the cardinality of the set $\{f_j^i\}$. This similarity measures the fraction of features of a template $i$, which has been uniquely matched to the set of test features.

### Matching Quality

In addition to the binary decision of whether a feature match is ambiguous - and thus ignored - or unique, introducing a measure of matching quality may improve the accuracy of a similarity measure and thus the performance of the identification system. Generally, the idea is to *weigh* the contribution of a feature match proportionally to its matching quality:

$$S(\{f_k\}^{test}, \{f_j\}^{temp})_i^{\text{QUALITY}} = \sum_k w_k \cdot vote(f_k^{test}, i), \tag{3.24}$$

where $w_k$ represents a quality measure. In appearance space, we can describe the *accuracy* of a feature match with respect to the appearance distance $d(f_1, f_2)$. As the accuracy decreases with increasing appearance distance, we define

$$w_k^{ACCURACY} = \frac{1}{d(f_k, f_k^{min})^\alpha}, \tag{3.25}$$

where $\alpha$ is a variable parameter. Motivated by the interpretation of the distance ratio $\frac{d(f_k, f_k^{min})}{d(f_k, f_k^{2nd})}$ as a measure of *ambiguity*, we define another quality function, taking the *confidence* of a feature match as the inverse ambiguity into

account:

$$w_k^{CONFIDENCE} = \frac{d(f_k, f_k^{2nd})}{d(f_k, f_k^{min})} \tag{3.26}$$

### Feature Distinctiveness

From a different perspective, we may define the *distinctiveness* of a feature $f_j^i$ for template $i$ as a function of all template features $\{\{f_j\}_i^{temp}\}$. We expect a feature $f_j^i$ to be distinctive for a template $i$ if its average minimum distance to features of other templates is high. According to our matching strategy, we take only features of other templates into account, which agree geometrically with the currently inspected feature. Average *minimum* distance means, that we compute for each template $k$ other than $i$ the minimum distance between the feature $f_j^i$ and all features of template $k$ and average these minimum distances. This distinctiveness may be evaluated in the learning stage and thus can be incorporated with no additional cost into the identification stage.

Eventually, any imaginable combination of these quality measures may increase the performance of an identification system. However, we focus on separately evaluating those defined above, in order to gain some insight into the particular impact each of them has on identification performance.

## 3.3 Combined Detection, Localization and Identification

The detector returns a set of OCI geometries. For each such geometry we have to decide individually, which subset of features extracted from the test image shall be used for identification. In this section we propose two rather contrary approaches. The *loosely coupled* system combination provides a large amount of test features to the identification system for maximal identification performance. This method addresses an application scenario in which a *general* face detector is used as a preprocessing step for identification. The *integrated recognition* model on the other hand, performs identification based on those detector model features, which supported the OCI hypothesis returned by the detector. With this rather *tight* system combination approach we intend to investigate exploitable synergies assuming a scenario in which the detection and

identification model are trained with images of the same subjects.

### 3.3.1 Loosely Coupled Detection and Identification

The template features for identification are stored in the database with their normalized geometry relative to the OCI (as described in Section 3.1.1). Therefore, we can easily determine a bounding box around a normalized OCI geometry, which encapsulates all template features. This bounding box in conjunction with a predicted OCI can be used to filter features extracted from a test image for identification. Note that with this definition of the bounding box, we keep *all* relevant information for identification, as features outside this bounding box would not agree geometrically with template features and thus would be unmatched anyway.

Consider a set of features $f_i$ together with their feature normalized relative OCI geometry $g_i$ and a set of geometrical thresholds $T^g : (T^{pos}, T^{scale}, T^{angle})$, which delimit the area of geometrical agreement (see Equation (3.8)). As a preprocessing step, we need to convert the geometries $g_i$, which currently represent a mapping from a normalized feature to the OCI, into a relative feature position to a normalized OCI. This procedure is described in detail in Section 3.1.1. Using the definition of geometrical agreement (Equation (3.8)), we can determine the normalized bounding box $BB_i : (x_i^{min}, y_i^{min}, x_i^{max}, y_i^{max})$ in horizontal and vertical direction about a feature position $g_i$ as follows:

$$\begin{aligned} x_i^{min} &= x_i - T^{pos} \cdot \sigma_i & \qquad y_i^{min} &= y_i - T^{pos} \cdot \sigma_i \\ x_i^{max} &= x_i + T^{pos} \cdot \sigma_i & \qquad y_i^{max} &= y_i + T^{pos} \cdot \sigma_i \end{aligned} \tag{3.27}$$

Based on the set of feature specific bounding boxes $\{BB_i\}$ corresponding to all template features, we can now determine the bounding box $BB\_MAX$, which includes all positions normalized to the OCI, which are relevant for identification:

$$\begin{aligned} x_{BB\_MAX}^{min} &= argmin_i(x_i^{min}) & \qquad y_{BB\_MAX}^{min} &= argmin_i(y_i^{min}) \\ x_{BB\_MAX}^{max} &= argmax_i(x_i^{max}) & \qquad y_{BB\_MAX}^{max} &= argmax_i(y_i^{max}) \end{aligned} \tag{3.28}$$

For scenarios in which faces are likely to overlap or for other contexts in which the maximal bounding box might not be the optimal choice regarding identification performance, we can define smaller bounding boxes in order to maximize the confidence with which the chosen features belong to the face corresponding to the OCI prediction. We specifically propose the average bounding box $BB_{AVG}$:

$$\begin{aligned} x^{min}_{BB\_AVG} &= \frac{1}{N} \cdot \sum_{i=1}^{N} (x_i^{min}) & y^{min}_{BB\_AVG} &= \frac{1}{N} \cdot \sum_{i=1}^{N} (y_i^{min}) \\ x^{max}_{BB\_AVG} &= \frac{1}{N} \cdot \sum_{i=1}^{N} (x_i^{max}) & y^{max}_{BB\_AVG} &= \frac{1}{N} \cdot \sum_{i=1}^{N} (y_i^{max}) \end{aligned} \tag{3.29}$$

For example, if faces in training images are not upright but the face region is marked parallel to the horizontal and vertical axes, this average bounding box is able to efficiently remove undesirable artifacts and features which are incidentally inside this face region, due to the limitations of defining it.

Now, consider a set of test features $\{f_k^{test}\}$ with geometries $g_k^{test}$ in absolute image dimensions. The detector localizes a face with OCI geometry $g_o$. Given a normalized bounding box $BB : (x^{min}, y^{min}, x^{max}, y^{max})$, how can we determine the subset of features of $\{f_k^{test}\}$ which are inside the bounding box $BB$ around the detected OCI geometry $o_g$? At first, we need to normalize the whole set of test features with respect to the OCI geometry $o_g$ as described in Section 3.1.1. Based on the normalized geometries, we can then determine the normalized feature location relative to the OCI. These locations can easily be compared to the boundaries defined by bounding box $BB$ and features located outside $BB$ may be discarded. At this point it would be more direct to normalize to $o_g$ instead of first normalizing the OCI to feature geometry and then determine the relative feature location, but we chose this description in order to keep with the geometrical relationships defined in Section 3.1.1.

### 3.3.2 Integrated Detection and Identification

Consider the case in which we are not interested in detecting general faces, but simply want to detect faces of individuals we aim to identify. In this scenario it might be sufficient or even beneficial with respect to identification accuracy, to

perform identification exclusively based on features extracted from a test image, which *match* detector model features.

Given a detected OCI geometry $g_o$, we propose to select the subset of features derived from a test image, which support the hypothesis of this particular OCI representing the presence of a face in the test image (see Section 3.1.3). This set of features is then used for identification as described in Section 3.2.

Independent of the identification performance achieved with this method, it has two great advantages over the loosely coupled model described above: For describing faces as sets of SIFT features (templates) in the database, it is no longer necessary to store the whole set of features extracted from a training image. As we use for identification only test features which match detector model features, all template features which do not agree with any detector model feature in terms of appearance and geometry may be discarded, which significantly reduces the total number of template features. Furthermore, the computational cost for recognition decreases, as filtering features within a bounding box is no longer necessary and more importantly the number of pairwise feature matches in the identification stage is reduced to a minimum.

# 4 Evaluation

In this chapter we present evaluation results for the detection model, our identification model and the combined recognition model. Before, we introduce some common performance measures and give some details on the data sets we use for evaluation.

## 4.1 Performance Metrics

The most basic quantities for describing the quality of a machine learning algorithm, on top of which more powerful performance measures will be defined, are the numbers of *true positive* responses (TP), *false positive* responses (FP), *true negative* responses (TN) and *false negative* responses (FN) of the system. For a classification problem, a positive response of a system is any response indicating that the inspected sample belongs to the class we are interested in. A face detector aims to find faces in images. Hence, any face found is considered as being a positive response. If at the specified location in the image really is a face, this response is true (TP), otherwise the response is false (FP). If a detector response indicates that at a specific location is no face present, its response is negative. Again, this is either true (TN) or false (FN). Note that the total number of possible false responses for a scale invariant face detector to a specific test image is undefined, because scale (in contrast to pixel coordinates) is not integer and possibly infinite, and even pixel coordinates may be defined on subpixel (real numbered) level.

These measures are combined to define higher classification statistics metrics:

$$\text{False positive rate:} \quad \frac{FP}{nF} \tag{4.1}$$

$$\text{True positive rate/ Recall:} \quad \frac{TP}{nP} \tag{4.2}$$

$$\text{Precision:} \quad \frac{TP}{TP + FP} \tag{4.3}$$

$$\text{1-Precision:} \quad \frac{FP}{TP + FP} \tag{4.4}$$

$$\tag{4.5}$$

The terms $nP$ and $nF$ denote the total number of positive samples and negative samples, respectively, in the evaluation data set.

In general, different classes (e.g. object categories) overlap in feature space or their perfect separation is unachievable due to model limitations. Thus, there is a trade-off between false alarms and misclassified positive samples. Many classification algorithms account for that with an adjustable parameter (e.g. an acceptance threshold). The quality of such an algorithm can therefore not be measured by a single value, but rather by a function describing this trade-off for any possible value of the adjustable parameter.

One method for describing this trade-off is the **Receiver Operating Characteristics** (ROC) curve. The ROC curve plots the true positive rate vs. the false positive rate (see Figure 4.1). Any ROC curve starts at the origin and ends at position $(1, 1)$. A classifier, which randomly[1] assigns a sample to any of two object categories, corresponds to the diagonal line in such a diagram. The ROC curve of an optimal classifier, which never falsely classifies any sample, follows the ordinate to the top left corner and then goes straight along the top of the diagram. Hence, the quality of a classification algorithm is considered to be better the more its ROC curve *approaches* the top left corner $(1, 0)$, which represents the highest possibly attainable accuracy.

As we have seen above, this method is not applicable to evaluate the face detector. In this context we are more interested in knowing how many of all faces in the evaluation set are correctly detected and how many of its detections are false.

A common way of expressing this relationship is plotting **Recall vs. 1-**

[1] assuming uniform distribution

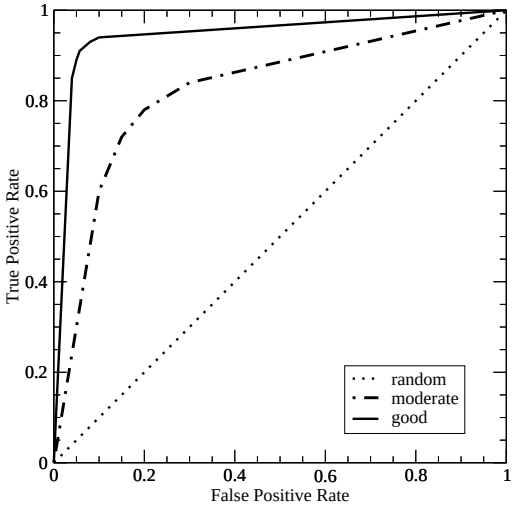


Figure 4.1: Example ROC curves: The performance of a classification system is considered better the more the curve approaches the point (1,0) representing no false classification

**Precision**. These plots allow for a similar graphical interpretation of the quality of a detection system: The more the curve approaches the top left corner, the better is the trade-off between correct detections and false alarms.

We will use these Recall vs. 1-Precision plots to describe the quality of our face detector. As described in Section 3.1.3, the prior ratio $\frac{p(face)}{p(bg)}$ may be used to influence the relative number of true vs. false detections. The performance of our identification system will be measured by the true positive rate, as the model contains no threshold, which explicitly models the trade-off between correct and false identifications on the face level (on feature level, the distance ratio threshold $T_r$ (Section 3.2.2) represents this trade-off to some extent).

## 4.2 The FERET Database

The FERET database has been assembled as part of the Facial Recognition Technology (FERET) program and is publicly available [PWHR98, PMRR00]. Its current version, the *Color FERET* database, has been released in October

2003[2]. The database contains 11338 facial color images of 994 subjects under varying viewing conditions and all images have a size of 512 by 768 pixels. Image files are in uncompressed PPM format. The collection of subjects is very diverse in terms of age and ethnicity. The photographs have been recorded in a total of 15 sessions over the course of three years. The images are neither aligned nor cropped. Thus, faces are of different size and at different locations in the image. Pictures are taken in front of a neutral background, but show a more or less large part of upper body clothing. Some subjects wear glasses or jewelery and some have facial hair.

The FERET program further defines an evaluation protocol to facilitate comparison of face recognition performance results reported by different authors. The entire set of images is arranged in *partitions*, representing training and evaluation sets for different face variations and viewing conditions. This section presents the partitions and the associated nomenclature, which we use for evaluation in Section 4.3.

The *training set/* gallery **fa** consists of one near frontal face image of each subject (994 images in total). The facial expression is not predetermined.

The partition **fb** consists of at most one other near frontal face image of each subject showing a different facial expression than in the corresponding image in **fa**(992 images in total).

All other near frontal face images of each subject are collected in the partition **dup1**. This set consists of a total of 736 images. The image set **dup2** is a subset of **dup1** containing 228 images taken at least 18 months later than the corresponding gallery image.

The images in the set **rb** show faces with a pose angle of 15° to the subject's right and includes 321 images. A quick reference for all of these partitions describing variation relative to the gallery set and number of images is provided in Table 4.1.

The predecessor of the Color FERET database is the *Gray FERET* database. As the name suggests, all images are gray level. They have a reduced resolution of 256 by 384 pixels and have undergone lossy compression. From Gray FERET to Color FERET, some subjects and the corresponding images as well as some partitions have been removed. We make particular use of the **fc** partition in our evaluation, which shows illumination variations to the gallery set.

[2]Official Homepage: http://face.nist.gov/colorferet

| Partition | #Images | Variation |
|---|---|---|
| **fa** | 994 | training set |
| **fb** | 992 | expression |
| **fc** | 194 | illumination |
| **dup1** | 736 | duplicates |
| **dup2** | 224 | aging (at least 18 months) |
| **rb** | 321 | pose 15° to subjects right |

Table 4.1: FERET image partitions used for evaluation

## 4.3 Evaluation Results

For our experiments we labeled all images used for training and testing with the OCI as a vector from the base of the nose to the forehead. Additionally, we manually defined for all images a paraxial rectangular *region of interest* (ROI) enclosing the face region as shown in Figure 4.2. This ROI is used for detection to separate face features from background features, and for identification to discard features corresponding to clothing. As subjects wear the same clothes in some test images as in the gallery, including these features artificially improves identification performance. All images of the Color FERET database have been converted to gray scale images. Besides that, we did not perform any normalization (e.g. histogram equalization, rotation, rasterization or accurate alignment). SIFT feature extraction is performed using David Lowe's demo software[3].

### 4.3.1 Detection

The detector model was trained with a subset of the Color FERET gallery set (**fa**). We constructed another subset of **fa** containing 500 images and subjects, which was used for testing. These sets did not share any subject. All features inside the ROI are used as face model features, and all remaining features are used as background features. The final clustering threshold $T^d$ for our agglomerative appearance clustering method was set to 0.5 (see Section 3.1.2). This threshold produced in our experiments the maximum number of appearance clusters consisting of at least two face model features, regardless the number of features used for training. The baseline geometry thresholds were set to $T^g : (T^{pos} = 0.5, T^\sigma = 1.5, T^\theta = \pi/2)$ as in [TA09]. Evaluation results are visu-

[3]Available at: http://www.cs.ubc.ca/ lowe/keypoints/

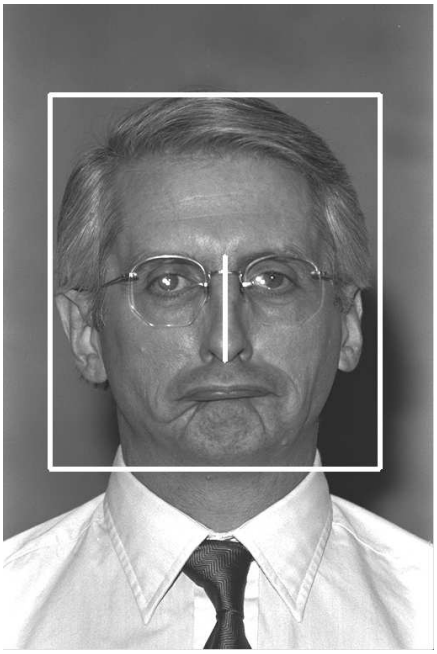

Figure 4.2: Example of a labeled image: We mark in each image a rectangular face region separating the face from the background and the OCI as a vector from the base of the nose to the forehead.

alized with Recall vs. 1-Precision plots (see Section 4.1). For better visibility, we plot only results for the top left quadrant, corresponding to ranges $[0.5, 1.0]$ for Recall and $[0, 0.5]$ for 1-Precision.

We consider a positive response (i.e. an OCI geometry $g_o$) as true positive, if $g_o$ agrees geometrically with the ground truth OCI $g_o^{gt}$, as defined in Section 3.1.1. For a series of OCI geometries $(g_o^1, \cdots, g_o^N)$ returned by the detector, which is in descending order of confidence $\gamma(H_i)$, we count only the first positive response as true positive if it agrees geometrically with $g_o^{gt}$, and all other positive responses regardless their geometrical agreement with $g_o^{gt}$ is considered as being false positive. Thus, duplicate correct responses are handled as false positives. Note that this method represents a *conservative* performance measure.

In order to determine the impact of the size of the training set on detector performance, we constructed sets of 200, 300, 400 and 500 images, respectively. Note that the number of face and background features does not grow linearly with the number of images as shown in Table 4.2. Evaluation for all models has been carried out on the same test set containing 500 images. Results are show in Figure 4.3. We can see that besides the low number of training features,

| **Images** | *face* **Features** | *bg* **Features** | *face* **Model Features** |
|---|---|---|---|
| 200 | 39687 | 102184 | 948 |
| 300 | 51792 | 130169 | 1074 |
| 400 | 66286 | 203833 | 1333 |
| 500 | 94488 | 243806 | 1615 |
| 994 | 214958 | 453019 | 2701 |

Table 4.2: Detector model training sets

even the smallest model consisting of less than 1000 features performs quite well. As we would expect, detection performance increases with the number of training features. The largest performance boost is obtained between 400 and 500 training images as with the former, the dent shared by the latter and weaker models at about 80% recall and 90% precision, disappeared. This might indicate, that the number of training features used to train the model from 500 images, is about the minimum number to obtain a *stable* face model with our method.

Towes and Arbel report a similar performance for their OCI based face detector in [TA06]. As they do not evaluate their detector on the FERET database but on a non-specified number of images from the Internet and do not state explicitly, how they handle correct duplicate detection, we can just assume that the performance of our detector is comparable. In [TA09], the authors report evaluation results on the CMU profile database, training the model as well with 500 images from the FERET database. In these more complex scenarios including multiple (partially occluded) faces per image and a high amount of background clutter, their detection model achieves about 40% recall at 18% precision. As the focus of this thesis lies on the combination of detection and identification, and large scale databases with complex scenes and *multiple* sample images per subject are to our knowledge unavailable to date, we leave this to future work.

Since images in the FERET database show only a single face, we measured the performance of the detector responding with a maximum of one OCI location in Figure 4.4. It can be seen that performance significantly increases by about 5% to 90% recall at 90% precision. This may be due to the fact that we count duplicate correct localizations as false positives.

With respect to a combined detection and identification system which is trained on the same data set, we train our detection model with the full FERET gallery set **fa**. Details on feature counts are presented in Table 4.2. For evalua-

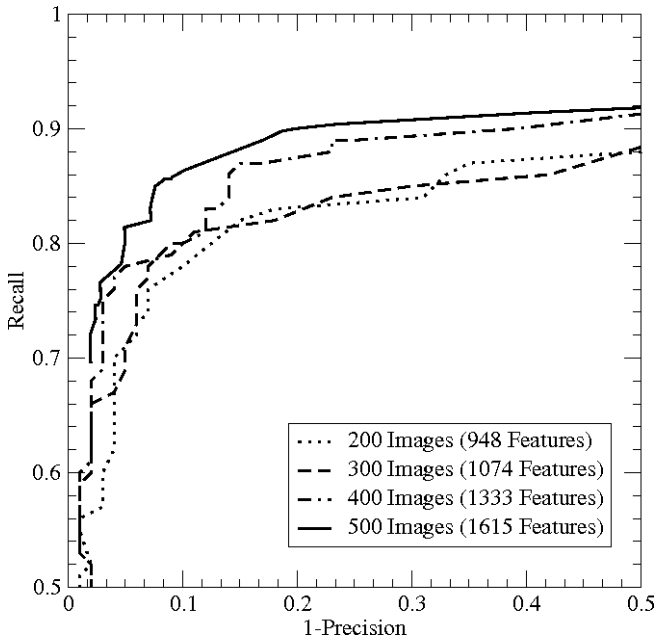


Figure 4.3: Detection performance with varying number of training images

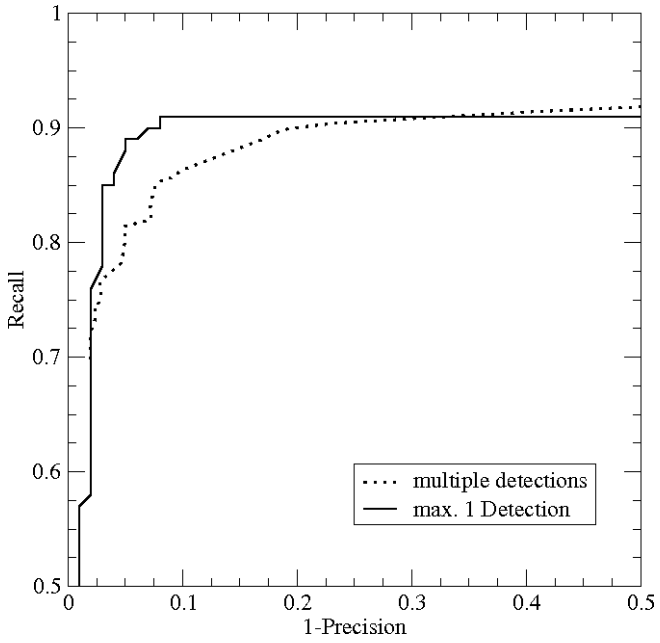


Figure 4.4: Detector performance depending on whether multiple responses or just a single detection response is allowed. Detector model **fa-500** tested the remaining images of **fa**

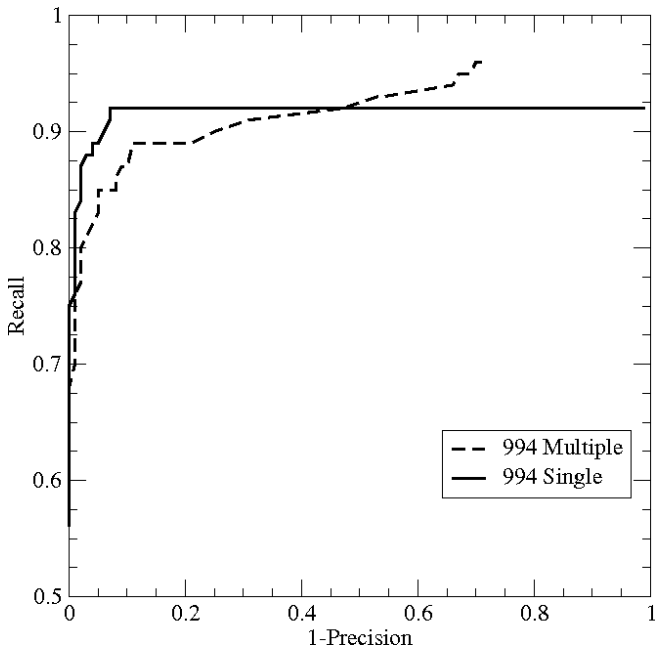


Figure 4.5: Detection model trained with the full gallery set **fa** and evaluated on **fb**.

tion, we used the test set on expression variations **fb**. Figure 4.5 shows evaluation results for the cases allowing multiple and just a single detector response. In the case of multiple responses, the we achieve 89% recall with a precision of 89 %. Allowing just a single detector response, the maximum recall of 92% is obtained with 93 % precision.

The results presented in this section show, that face detection and localization based on SIFT features with our method in these rather simple scenarios is accomplished with high accuracy. The detection model trained with 500 images is capable of correctly detecting 85-90% of all test images with more than 90% precision. A correct OCI is on average detected based on less than ten SIFT features. An example detection result showing the detected OCI and the image features supporting this particular hypothesis is shown in Fig. 4.6. In the following section we present the evaluation results on our identification model before combining them to a full face recognition model.

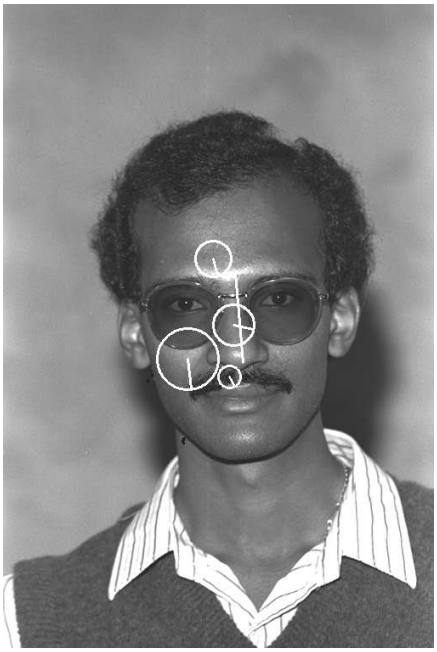

Figure 4.6: Example detection: The image shows a correctly detected OCI and the features supporting this OCI hypothesis. In this case the detector correctly predicted the OCI location based on four SIFT features

### 4.3.2 Identification

We test the identification model using the labeled ROI and OCI in training and test images. All features of a training image inside the ROI are normalized to the OCI geometry (see Section 3.1.1) and stored in the database. From any test image, we extract solely features inside the ROI and normalize their geometries to the OCI. Then, we compute the similarity measures between the set of test features and all templates, and consider the template with the highest similarity as being present in the image. We start by using the simple voting scheme (introduced in Section 3.2.2) to compute similarity measures.

With this measure of similarity, the only variable parameters are the distance ratio threshold $T_r$, which *decides* which feature matches are unique enough to vote for a particular subject, and the geometrical thresholds $T^g : (T^{pos}, T^\sigma, T^\theta)$, which define the range of geometries considered to agree with a particular geometry in 2D position, scale and angle. For training, we used the full gallery set **fa**. Evaluation on illumination variations (**fc**) is carried out on the Gray FERET database. In this case we did not use the full gallery set **fa**, but only

| | **fb** | **fc** | **dup1** | **dup2** | **rb** | **Total** |
|---|---|---|---|---|---|---|
| $T_r = 0.6$ | 95.87 | 59.79 | 44.29 | 19.74 | 38.63 | 63.21 |
| $T_r = 0.7$ | 97.28 | 79.38 | 50.04 | 30.70 | 47.04 | 69.13 |
| $T_r = 0.8$ | 97.56 | 87.63 | 56.79 | 44.74 | 51.75 | 73.81 |
| $T_r = 0.9$ | 98.08 | 94.85 | 60.60 | 49.56 | 59.81 | 77.21 |
| $T_r = 1.0$ | **98.89** | **97.42** | **60.87** | **54.39** | **66.67** | **79.15** |
| **Min+2nd** | 98.89 | 97.42 | 60.73 | 52.19 | 66.98 | 78.95 |
| **Min**$+(1.25 \cdot d(Min))$ | 97.08 | 92.78 | 54.35 | 42.98 | 61.99 | 74.46 |

Table 4.3: Impact of distance ratio threshold $T_r$ on identification performance (true positive rate in %)

the subset containing individuals included in the test set **fc** (194 individuals with one image per subject). Besides evaluation results on particular test sets, we present a total identification performance. This total performance is the average performance on all test sets, where the contribution of the result on a particular test set is weighted by the number of images it contains.

The effects of the distance ratio threshold on identification performance are shown in Table 4.3. The results show a monotonic increase of identification performance with growing threshold values. Note that a threshold $T_r = 1$ corresponds to accepting any best feature match regardless the appearance distance to the second best match. This may be caused by the fact that the appearance variance of a SIFT feature between several images of the same face overlaps with the appearance variance across individuals. Based on these results, we tested some additional similarity measures, letting a test feature not only vote for a single but for multiple templates in the database. We specifically explored the performance of a similarity measure *Min+2nd*, where a feature votes for the template which best matches the test feature as well as for the second best feature match. In the similarity measure $Min+(x \cdot d(Min))$, a feature votes for any template, for which the distance of the best feature match is smaller than $x$ times the distance to the absolute best feature match. As shown in Table 4.3 though, these similarity measures did not yield a higher identification performance.

These first results also give an insight into the difficulties involved with different viewing conditions. Expression variations seem to be the most simple kind of variations. Our identification system shows for these variations with respect to changing distance ratio thresholds a high stability in identification

| $T_{pos}$ | $T_\sigma$ | $T_\theta$ | **fb** | **fc** | **dup1** | **dup2** | **rb** | **Total** |
|---|---|---|---|---|---|---|---|---|
| 0.5 | 1.5 | $\pi$ / 6 | 98.89 | 97.42 | 60.87 | 54.39 | 66.67 | **79.15** |
| [0.3] | 1.5 | $\pi$ / 6 | 98.08 | 93.81 | 61.28 | 55.26 | 63.86 | 78.39 |
| [0.7] | 1.5 | $\pi$ / 6 | 98.89 | 96.39 | 60.73 | 52.19 | 65.42 | 78.59 |
| 0.5 | [1.3] | $\pi$ / 6 | 98.89 | 95.88 | 61.55 | 53.95 | 66.66 | **79.20** |
| 0.5 | [1.7] | $\pi$ / 6 | 98.89 | 97.42 | 60.33 | 53.51 | 66.36 | 78.88 |
| 0.5 | 1.5 | [$\pi$ / 8] | 98.79 | 97.42 | 60.46 | 53.95 | 66.46 | 78.92 |
| 0.5 | 1.5 | [$\pi$ / 4] | 98.89 | 96.91 | 60.46 | 53.93 | 96.91 | 78.71 |

Table 4.4: Sensitivity of identification performance (true positive rate in %) to changes in the geometrical thresholds. Thresholds differing from the baseline parameters in the first row are put in square brackets.

performance. Results on illumination variations suggest a similar level of difficulty, although identification performance strongly varies with changes in the distance threshold. The effects of *aging* on face appearance seem to be rather challenging, as identification performance on the data sets **dup1** and especially **dup2** is comparably low. The moderate performance on face rotations in depth (**rb**) might be due to a high sensitivity of SIFT descriptors to these variations, but also to our geometry based potential matching feature selection method.

Therefore, we systematically evaluate the sensitivity of our face identification model to changes in the geometrical thresholds. Table 4.4 shows the results obtained by changing either of the thresholds $T^{pos}, T^\sigma$ and $T^\theta$, corresponding to 2D position, scale and angle, respectively. The first row shows the results for our *baseline* thresholds, and for all other experiments the changed threshold is put in square brackets for better readability. Based on these evaluations, we decide to keep the *baseline* thresholds for future experiments. Although a scale threshold of $T^\sigma = 1.3$ yields a slightly better identification performance in total, this low threshold is expected to be too restrictive for our detection model and thus the combination of both models.

### Comparison of Similarity Measures

In Section 3.2.2 we proposed a rather wide range of similarity measures incorporating various factors such as the number of template features per individual (NORMALIZED), the appearance distance between test and template feature (ACCURACY), the distance ratio of best match vs. second best match (CONFIDENCE) and the template feature distinctiveness, defined as the average

| | fb | fc | dup1 | dup2 | rb | Total |
|---|---|---|---|---|---|---|
| **VOTING** | 98.89 | 97.42 | 60.87 | 54.39 | 66.67 | **79.15** |
| **NORMALIZED** | 98.89 | 97.42 | 53.40 | 35.96 | 54.52 | 73.65 |
| **ACCURACY,** $\alpha = 1/2$ | 98.89 | 97.94 | 59.92 | 52.63 | 61.99 | 78.15 |
| **ACCURACY,** $\alpha = 1$ | 98.69 | 96.90 | 57.47 | 47.37 | 58.88 | 76.36 |
| **ACCURACY,** $\alpha = 2$ | 93.55 | 87.11 | 42.57 | 18.85 | 38.94 | 63.87 |
| **CONFIDENCE** | 98.89 | 97.94 | 60.46 | 53.07 | 66.38 | 78.92 |
| **DISTINCTIVENESS** | 98.67 | 95.36 | 59.79 | 51.32 | 66.36 | 78.26 |

Table 4.5: Identification performance (true positive rate in %) using various similarity measures.

minimum distance to features of other templates (DISTINCTIVENESS). The evaluation results of our identification model using these similarity measures are presented in Table 4.5. Surprisingly, none of these similarity measures beats the performance of the simple voting scheme. Normalizing the similarity measure by the number of template features might be disadvantageous, because for each template a variable fraction of features stored in the database might be unsuitable for uniquely identifying a particular individual. Either because a feature is not distinct enough, or because it is unlikely to appear in other images of the same subject (e.g. background clutter). The matching accuracy as defined might not actually be an appropriate measure to express the quality of a feature match. The same might be the case for feature distinctiveness. Incorporating the distance ratio into the similarity estimation yields promising results, although the linear contribution of this quality measure did not improve identification performance.

### Comparison to Other Identification Models

In the literature, many face identification methods have been evaluated on the FERET database. Thus, we choose to evaluate our model on this particular database, in order to compare our performance results with those reported by other authors.

Table 4.6 compares our identification results (*SIFT_OCI*) with other SIFT based identification methods (*SIFT_GRID* and *SIFT_CLUSTER*), the local feature based methods *Elastic Bunch Graph Matching* (EBGM) and *Local Binary Patterns* (LBP) and the holistic *Fisherface* approach. The evaluation results for all methods (except ours) are taken from [LMT+07]. For the evaluation

| Methods | fb | fc | dup1 | dup2 | Total |
|---|---|---|---|---|---|
| Fisherface [BHK97] | 94 | 73 | 55 | 31 | 72 |
| EBGM [WFKvdM97] | 90 | 42 | 46 | 24 | 64 |
| LBP [AHP04] | 97 | 79 | **66** | **64** | **81** |
| SIFT_GRID [BLGT06] | 94 | 35 | 53 | 36 | 69 |
| SIFT_CLUSTER [LMT+07] | 97 | 47 | **61** | 53 | 76 |
| SIFT_OCI | **99** | **92** | **61** | **54** | **81** |

Table 4.6: Comparison of the performance (true positive rate in %) of our face identification model (SIFT_OCI) with other methods for face identification.

of those methods, all images have been normalized (using histogram equalization [AR05]), aligned (both centers of the eyes are on a common line), cropped (showing only the face) and rasterized (scaled to a fix resolution). Note that the authors used the original Gray FERET database for all experiments, while we use - except for experiments on illumination variations, the Color FERET database for evaluation. However, as there is little difference between the Color FERET and the Gray FERET database (see Section 4.2), we consider the evaluation results as comparable. While we used above for the evaluation of illumination variations (partition **fc** of the Gray FERET database) only the subset of the gallery **fa** corresponding to subjects in **fc**, we now use the full gallery **fa** to construct our database for better comparison.

The results in Table 4.6 show, that our identification model is very competitive. While LBP performs better on the test sets **dup1** and **dup2**, the proposed method shows a better identification performance on expression variations (**fb**) and achieves outstanding results on illumination variations **fc**. Additionally, our identification model outperforms both of the other SIFT based methods.

### 4.3.3 Recognition - Combined Detection and Identification

We evaluate the combined recognition model using two different sets of model features for the face detector: The set **fa-full** contains all detector model features derived from model training (Section 3.1.2) with the whole FERET gallery set **fa**. The face model set **fa-500** contains all model features learned from a subset of 500 images of **fa**. In combination with the detection model **fa-full**, we use the gallery set **fa** to create the database of templates for identification.

Thus, detection and identification models are *based on the same set* of features. In combination with the detector model feature set **fa-500**, we construct the database for identification from the remaining images of **fa**, which have not been used to train the detector model **fa-500**. As opposed to the former configuration, this latter setting represents a scenario, in which we use a *general face detector* (not particularly trained to detect the subjects we want to identify) as a preprocessing module for identification.

There are several performance criteria we measure in our experiments: We consider a face which has been correctly detected and correctly identified as a true positive response of the combined system. As all subjects in the test set are known to the identification system, the total number of possible positive responses equals the number of faces in the test set. We denote the fraction of correctly detected and correctly identified faces of the total number of faces in the test set as **ID/Exp.**. Additionally, we measure the fraction of correctly detected *and* correctly identified faces of all correctly *detected* faces, which we denote as **ID/Det.**. This measure indicates the portion of the overall system performance achieved by the identification system. We continue to describe the detector performance in terms of recall and precision. Note that for the evaluation of the combined system, the acceptance threshold of the detector is set to a fixed value, and thus the detector performance is described by a pair of values (recall and precision for a specific threshold) rather than a curve as in Section 4.3.1.

#### Loose Coupling

The loosely coupled system combination approach as introduced in Section 3.3.1 uses the OCI returned by the detector to define a bounding box and selects all features from the test image, which are situated inside this bounding box, for identification. We proposed two types of bounding boxes - namely *BB_MAX* and *BB_AVG*.

**Bounding Boxes:** In a first comparative evaluation we attempt to determine the type of bounding box, which achieves the better performance, using the performance criteria introduced above. For evaluation, we used the model combinations defined above: detector model **fa-full**, for which both subsystems are trained with the data set **fa**, and the detector model **fa-500**, which is combined

| Bounding Box | Model | ID/ Det. | ID/ Exp. |
|---|---|---|---|
| BB_MAX | fa-500 | 89.98 | 80.8 |
| BB_AVG | fa-500 | 98.44 | 88.4 |
| BB_MAX | fa-full | 93.39 | 85.48 |
| BB_AVG | fa-full | 98.5 | 87.2 |

Table 4.7: Performance of the combined system using different types of bounding boxes, evaluated on two different detector models.

with an identification model trained with the subset of images of **fa**, which have not been used for detector training. For testing the model **fa-full**, we used the whole set **fb**, while the model **fa-500** was tested on the subset of **fb** corresponding to the subjects, which are known to the identification system. The results of these experiments are presented in Table 4.7. As the choice of bounding box does not influence the detector performance, we did not present precision and recall of the detector.

The results clearly show, that the average bounding box *BB_AVG* of template features is better suited for our system combination than the maximum bounding box *BB_MAX* enclosing all features of the database. This may be due to the fact that the maximum bounding box is too wide and thus includes too many background features for identification. Manually labeled rectangular face regions (ROI) are not defined relative to the labeled OCI's orientation, but along the horizontal and vertical axes of the image. Thus, if in a training image the relative orientation of the OCI to the y-axis approaches 45°, the resulting bounding box dimensions represent the diagonal of the face[4], and not its width and its hight. The average bounding box is more robust to this kind of artifacts. Therefore, we use the bounding box *BB_AVG* for all other experiments.

**Impact of Detector Acceptance Threshold on Identification Performance:** In a second experiment, we explore the impact of the detector acceptance threshold on the identification performance of the combined system. Recall that identification performance is described as the fraction of correctly detected and correctly identified faces of all correctly detected faces (ID/Det.). We use the same configurations of training and test sets as above and the average bounding box *BB_AVG* for feature selection. For each model, we choose an acceptance thresh-

[4]assuming a squared bounding box for simplification

| Model | Precision | Recall | ID/ Det. |
|---|---|---|---|
| fa-500 | 24.5 | 94.5 | 98.5 |
| fa-500 | 88.2 | 86.4 | 98.6 |
| fa-full | 29.2 | 96.0 | 98.3 |
| fa-full | 88.9 | 88.5 | 98.5 |

Table 4.8: Performance of the combined system using different acceptance thresholds for detection

old value with very low precision and the threshold value corresponding to the optimal trade-off between false positive and false negative detector responses. The results presented in Table 4.8 show, that the identification performance on correctly detected faces does practically not depend on this variable parameter of the detection system. If the acceptance threshold had an effect on the identification performance in the way that faces detected with high confidence are more likely to be correctly identified, this would indicate that faces which are *easier* to detect would also be easier to identify based on this detection. Either because faces showing features used for detection are more likely to be identified, or because the OCI localization of a face detected with high confidence is more accurate.

**Localization Accuracy:** In order to evaluate the accuracy of the OCI localization, we compare the identification performance of detected faces using the detected OCI with the identification performance of these faces using the labeled *ground truth* OCI (Table 4.9). If an OCI was badly localized, corresponding features between two images would not agree geometrically and thus would not be matched in the identification stage (as described in Section 3.2.1. Training sets, test sets, the bounding box type and the detector acceptance thresholds are chosen as above. The results are somewhat contradictory. For the model **fa-500**, the performance (ID/Det.) of identification based on the labeled OCI is slightly worse than using the OCI predicted by the detector. This may indicate that the predicted OCI defines a better geometrical correspondence between template features and features in the test image. On the other hand, using the model **fa-full**, we observe that the predicted OCI performs worse than its manually labeled counterpart, as we would expect. However, the performance differences between identification based on the labeled OCIs and predicted OCIs are so small, that we may conclude that the detector localizes an OCI with high

| Model | Precision | Recall | Detected OCI | Ground Truth OCI |
|---|---|---|---|---|
| fa-500 | 24.5 | 94.5 | 98.5 | 98.3 |
| fa-500 | 88.2 | 86.4 | 98.6 | 98.4 |
| fa-full | 29.2 | 96.0 | 98.3 | 98.8 |
| fa-full | 88.9 | 88.5 | 98.5 | 99.0 |

Table 4.9: Comparison of identification performance (ID/Det.) based on the detected OCI with identification based on labeled ground truth OCI

| Test Set | Precision | Recall | ID/Det | ID/ Exp. |
|---|---|---|---|---|
| fb | 90.71 | 89.80 | 98.44 | 88.40 |
| dup1 | 83.37 | 86.37 | 70.67 | 61.04 |
| dup2 | 84.76 | 84.76 | 56.18 | 47.62 |
| rb | 66.78 | 65.26 | 58.21 | 37.99 |

Table 4.10: Performance (in %) of the **loosely coupled** recognition model under various viewing conditions (detection model **fa-500**)

accuracy.

While we focused in the experiments above on the identification performance in case a face has been correctly detected, we now evaluate the performance of the combined recognition system under various viewing conditions. Therefore, we choose the detector acceptance threshold, detector models and identification gallery images as above. We test this system configuration for the model **fa-full** on the whole test sets **fb**, **dup1**, **dup2** and **rb**. In combination with the detector model **fa-500**, we use for evaluation the subsets corresponding to the images of the gallery set, which have not been used for detector model training and form the database for the identification system. The evaluation results are presented in Tables 4.11 and 4.10. With respect to detector performance, we see that precision and racall highly depend on the test set. The identification performance (ID/Det.) is comparable to the results achieved with the stand-alone face identification system. Although the model **fa-500** shows a better identification performance on the **dup1** test set, note that these results are not comparable as this combined system is only evaluated on a subset of **dup1**. By comparing recall, the fraction of correctly identified to correctly detected faces and the fraction of correctly identified faces to all experiments, we see that the lower recognition rate is almost exclusively due to the detection rate. Thus, the loose system combination approach may be considered as being very effective,

| Test Set | Precision | Recall | ID/Det. | ID/ Exp. |
|---|---|---|---|---|
| fb | 92.84 | 91.53 | 98.35 | 90.02 |
| dup1 | 89.79 | 87.64 | 59.07 | 51.77 |
| dup2 | 84.07 | 83.33 | 51.59 | 42.98 |
| rb | 76.07 | 72.27 | 66.38 | 47.98 |

Table 4.11: Performance (in %) of the **loosely coupled** recognition model under various viewing conditions (detection model **fa-full**)

as the performance decrease in comparison with pure identification is not caused by the system combination but by the rather poor detection performance.

### Integrated Detection and Identification

The *tight* system combination approach as described in Section 3.3.2 uses only those test features for identification, which correspond to detector model features and *support* the detected OCI hypothesis. Therefore, the detector needs to be slightly modified. In addition to returning a single or multiple OCIs, the detector now returns for each detected OCI the corresponding test features. As depicted in Figure 4.6, the number of these features is rather small. Therefore it is rather optimistic to image that this model can accurately identify a subject out of more than 500 subjects in the database, based on 5-10 features. Nevertheless, we evaluated this approach to gain some insight into the distinctiveness of SIFT descriptors. In this context we performed two series of experiments.

The first (results in Table 4.12) uses the **fa-full** detection model and is evaluated on the whole test sets **fb**, **dup1**, **dup2** and **rb**. The second series of experiments was conducted using the **fa-500** detection model and was evaluated on the subsets of **fb**, **dup1**, **dup2** and **rb**, which do not contain any subject which has been involved in detector training. These results are shown in Table 4.12. In general, these results are not as bad as expected: this tightly coupled approach using only an extremely small amount of features for identification, still identifies about 70% of the detected subjects of test set **fb** correctly! On the other test sets however, the identification performance (ID/Det.) and especially the recognition performance (ID/Exp.) are very low. Another meaningfull observation is that the **fa-500** model, which does not use the same subjects for training the detector and identification, performs significantly better. Note that although the results are not directly comparable as the test sets are different,

| Test Set | Precision | Recall | ID/Det | ID/ Exp. |
|---|---|---|---|---|
| fb | 92.84 | 91.53 | 69.93 | 64.01 |
| dup1 | 89.21 | 87.64 | 19.38 | 16.98 |
| dup2 | 84.07 | 83.33 | 7.37 | 6.14 |
| rb | 76.07 | 72.27 | 19.40 | 14.02 |

Table 4.12: Performance (in %) of the **integrated** recognition model under various viewing conditions (detection model **fa-full**)

| Test Set | Precision | Recall | ID/Det | ID/ Exp. |
|---|---|---|---|---|
| fb | 90.71 | 89.80 | 69.49 | 62.4 |
| dup1 | 83.37 | 86.37 | 28.00 | 24.18 |
| dup2 | 84.76 | 84.76 | 13.48 | 11.43 |
| rb | 66.78 | 65.26 | 22.31 | 17.53 |

Table 4.13: Performance (in %) of the **integrated** recognition model under various viewing conditions (detection model **fa-500**)

this may indicate that a *general* face detector - trained on arbitrary faces - is better suited to be used in combination with this integrated recognition scheme. A possible explanation for this observation is the following: Model features trained for detection are supported by at least two features of the training set. If we use the *same* training set for identification, and the detector model features are chosen to be non-distinctive for a particular subject in this training set, it is rather unsurprising that this approach may yield a lower recognition rate.

# 5 Conclusion

In this diploma thesis, we designed and implemented a full face recognition system based on SIFT descriptors and proved the feasibility of this approach.

We adapted the learning procedure of an existing face detection model using an agglomerative clustering approach, which requires less computation time. The experiments carried out on the FERET database showed that this model achieves high detection rates localizing a face with high accuracy and with a low number of false detections under these rather constraint conditions. Evaluation on more complex data sets may be necessary to further explore the potential of this detection model.

For face identification, we propose a new model using SIFT descriptors and the Object Class Invariant. We developed a new feature matching strategy based on this OCI, which further reduces the number of potential feature matches. This feature selection strategy does not only incorporate the 2D image position but also the scale and the angle of a feature, which further reduces the number of false correspondences. We systematically evaluated various similarity measures, with the surprising result that the simple voting scheme achieves the best identification rates. The proposed face identification method showed to be competitive with other face identification models such as SIFT-Cluster and Local Binary Patterns. Furthermore, the OCI based identification model does not require preprocessing steps such as histogram equalization or rasterization. Labeling the OCI in an image is sufficient to identify faces invariant to scale and in-plane rotation.

For the combination of face detection and identification, we experimented with two different approaches: The loose coupling technique uses OCI information provided by the detector to define a scale and orientation invariant bounding box, based on which features for face identification are selected. This technique does not additionally decrease identification performance and may thus be considered as being well suited for the combination of the proposed models.

The integrated detection and identification model uses only a very small number of features for identification. However, this approach showed an adequate recognition performance under variation of facial expression.

# Bibliography


[AHP04] Timo Ahonen, Abdenour Hadid, and Matti Pietikäinen. Face recognition with local binary patterns. *Lecture Notes in Computer Science: Proceedings of the 8th European Conference on Computer Vision, Prague, Czech Republic*, 3021:469–481, 2004.

[AR05] Tinku Acharya and Ajoy K. Ray. *Image Processing: Principles and Applications*. Wiley-Interscience, New Jersey, USA, 2005.

[BBTD03] David S. Bolme, J. Ross Beveridge, Marcio Teixeira, and Bruce A. Draper. The csu face identification evaluation system: Its purpose, features, and structure. *Third International Conference on Computer Vision Systems, Graz, Austria*, 2626, 2003.

[BHK97] Peter N. Belhumeur, João P. Hespanha, and David J. Kriegman. Eigenfaces vs. Fisherfaces: Recognition using class specific linear projection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19:7:711–720, 1997.

[BLGT06] Manuele Bicego, Andrea Lagorio, Enrico Grosso, and Massimo Tistarelli. On the use of SIFT features for face authentication. *IEEE International Conference on Computer Vision and Pattern Recognition Workshop, New York City, USA*, 2006.

[BTG06] Herbert Bay, Tinne Tuytelaars, and Luc Van Gool. SURF: Speeded up robust features. *European Conference On Computer Vision, Graz, Austria*, 2006.

[CL06] Gustavo Carneiro and David Lowe. Sparse flexible models of local features. *European Conference On Computer Vision, Graz, Austria*, pages 29–43, 2006.

[CSM08] Claudio Cruz, Enrique Sugar, and Eduardo F. Morales. Real-time face recognition for human-robot interaction. *IEEE International Conference on Automatic Face and Gesture Recognition, Amsterdam, The Netherlands*, 2008.

[Dau88] J. G. Daugman. Complete discrete 2-d gabor transforms by neural networks for image analysis and compression. *IEEE Transactions on Acoustics, Speech and Signal Processing*, 36 (7), 1988.

[DSC09] Geng Du, Fei Su, and Anni Cai. Face recognition using SURF features. *SPIE The Sixth International Symposium on Multispectral Image Processing and Pattern Recognition,*, 7496, 2009.

[DSHN09] Philippe Dreuw, Pascal Steingrube, Harald Hanselmann, and Hermann Ney. SURF-Face: Face recognition under viewpoint consistency constraints. *British Machine Vision Conference, London*, 2009.

[FB81] Martin A. Fischler and Robert C. Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24:381–395, 1981.

[FFFP03] Li Fei-Fei, Rob Fergus, and Pietro Perona. A bayesian approach to unsupervised one-shot learning of object categories. *IEEE International Conference on Computer Vision, Nice, France*, 2:1134–1141, 2003.

[FH75] K. Fukunaga and L. Hostetler. The estimation of the gradient of a density function, with applications in pattern recognition. *IEEE Transactions on Information Theory*, 21:32–40, 1975.

[Fis36] Ronald Aylmer Fisher. The use of multiple measurements in taxonomic problems. *Annals of Eugenics*, 7:179–188, 1936.

[FS95] Yoav Freund and Robert E. Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. *European Conference on Computational Learning Theory, Barcelona, Spain*, pages 23–37, 1995.

[Hje01] Erik Hjelmås. Face detection: A survey. *Computer Vision and Image Understanding*, 83:236–274, 2001.

[KS04] Yan Ke and Rahul Sukthankar. PCA-SIFT: A more distinctive representation for local image descriptors. *IEEE International Conference on Computer Vision and Pattern Recognition, Washington D.C., USA*, 2:506–513, 2004.

[LMT+07] Jun Luo, Yong Ma, Erina Takikawa, Shihong Lao, Masato Kawade, and Bao-Liang Lu. Person-specific SIFT features for face recognition. *IEEE International Conference on Acoustics, Speech and Signal Processing, Honolulu, Hawaii, USA*, 2:593–596, 2007.

[Low01] David G. Lowe. Local feature view clustering for 3D object recognition. *IEEE International Conference on Computer Vision and Pattern Recognition, Kauai Marriott, Hawaii, USA*, 1, 2001.

[Low04] David G. Lowe. Distinct image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60:2:91–110, 2004.

[Mac67] J. MacQueen. Some methods for classification and analysis of multivariate observations. *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability*, 1:281–297, 1967.

[MLS06] Krystian Mikolajczyk, Bastian Leibe, and Bernt Schiele. Multiple object class detection with a generative model. *IEEE International Conference on Computer Vision and Pattern Recognition, New York City, USA*, pages 26–36, 2006.

[MMP04] Pierre Moreels, Michael Maire, and Pietro Perona. Recognition by probabilistic hypothesis construction. *European Conference On Computer Vision, Prague, Czech Republic*, pages 55–68, 2004.

[MS05] Krystian Mikolajczyk and Cordelia Schmid. A performance evaluation of local descriptors. *IEEE Transactions On Pattern Analysis and Machine Intelligence*, 27:1615–1630, 2005.

[OFG97] E. Osuna, R. Freund, and F. Girosi. Training support vector machines: An application to face detection. *IEEE International Conference on Computer Vision and Pattern Recognition, San Juan, Puerto Rico, USA*, 6, 1997.

[PL00] Arthur R. Pope and David G. Lowe. Probabilistic models of appearance for 3D object recognition. *International Journal of Computer Vision*, 40:2:149–167, 2000.

[PMRR00] P.J. Philipps, H. Moon, S.A. Rizvi, and P.J. Rauss. The FERET evaluation methodology for face recognition algorithms. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22:1090–1104, 2000.

[PWHR98] P.J. Phillips, H. Wechsler, J. Huang, and P. Rauss. The FERET database and evaluation procedure for face recognition algorithms. *Image and Vision Computing*, 16:5:295–306, 1998.

[RBK98] H. A. Rowley, S. Baluja, and T. Kanade. Neural network-based face detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20:23–38, 1998.

[RK+07] A. Rattani, D. R. Kisku, , M. Bicego, and M. Tistarelli. Feature level fusion of face and fingerprint biometrics. *First IEEE International Conference on Biometrics: Theory, Applications, and Systems, Washington D.C., USA*, pages 1–6, 2007.

[SBB02] T. Sim, S. Baker, and M. Bsat. The CMU pose, illumination, and expression (PIE) database. *Proceedings of the IEEE Interenational Conference on Automatic Face and Gesture Recognition, Washington D.C., USA*, 2002.

[Sch99] Cornelia Schmid. A structured probabilistic model for recognition. *IEEE Conference on Computer Vision and Pattern Recognition, Fort Collins, USA*, pages 485–490, 1999.

[SK87] L. Sirovich and M. Kirby. Low-dimensional procedure for the characterization of human faces. *Journal of the Optical Society of America*, A 4:519524, 1987.

[TA06] Matthew Toews and Tal Arbel. Detection over viewpoint via the object class invariant. *IEEE International Conference on Pattern Recognition, Quebec City, Canada*, 1:765–768, 2006.

[TA07] Matthew Toews and Tal Arbel. Detecting, localizing and classifying visual traits from arbitrary viewpoints using probabilistic local feature modeling. *IEEE Workshop Analysis and Modeling of Faces and Gestures, Rio de Janeiro, Brazil*, pages 154–167, 2007.

[TA09] Matthew Toews and Tal Arbel. Detection, localization, and sex classification of faces from arbitrary viewpoints and under occlusion. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31:1567–1581, 2009.

[VJ01] Paul Viola and Michael Jones. Rapid object detection using a boosted cascade of simple features. *IEEE International Conference on Computer Vision and Pattern Recognition, Kauai Marriott, Hawaii, USA*, pages 511–518, 2001.

[WFKvdM97] Laurenz Wiskoktt, Jean-Marc Fellous, Norbert Krüger, and Christoph von der Malsburg. Face recognition by elastic bunch graph matching. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19:7:775–779, 1997.

[ZCRP03] W. Zhao, R. Chellappa, A. Rosenfeld, and P.J. Phillips. Face recognition: A literature survey. *ACM Computing Surveys*, pages 399–458, 2003.