Locating Landmarks Using Templates

Dissertation

zur Erlangung des akademischen Grades eines

Doktors der Naturwissenschaften

(Dr. rer. nat.)

dem Fachbereich Mathematik der Universität Duisburg-Essen vorgelegt

im August 2006

von

Jan Kalina, geb. in Prag (Tschechische Republik)

Tag der mündlichen Prüfung:	12. Januar 2007
Gutachter:	Prof. Dr. P.L. Davies (Universität Duisburg-Essen)
	Prof. Dr. Daniel Peña (Universidad Carlos III de Madrid)

Contents

List of Figures

List of Tables

Chapter 1

Introduction. Motivation.

1.1 Introduction

The Institute of Human Genetics is working on interesting problems in the genetic research using images of faces. The ambitions of the research are to classify automatically genetic syndromes from a picture of a face; to examine the connection between the genetic code and the size and shape of facial features; and also to visualize a face based only on its biometric measures. Some of the results are described in the paper by Loos et al. (2003). There are images of 55 patients considered and each of them can be classified to one of five groups according to a genetic malformation deforming the face. The correct classification rate of the syndromes by an automatic procedure based only on the image of the face was 84 % which is considered remarkably successful.

The paper by Böhringer et al. (2005) presents a larger study. There are patients classified to one of 10 different syndromes and on average there are about 12 individuals present in each group. The paper again tries to recognize the syndrome in each person. For different syndromes the success rate lies between 75 % and 80 %.

Locating the landmarks is always the first step of all such procedures, however not the primary goal of the study. The landmarks are prominent parts of the face. Let us consider one of the images coming from Institute, which is shown in Figure 1.1. Figure 1.2 shows some

Figure 1.1: An example of an image: the photo of dr. Stefan Böhringer.

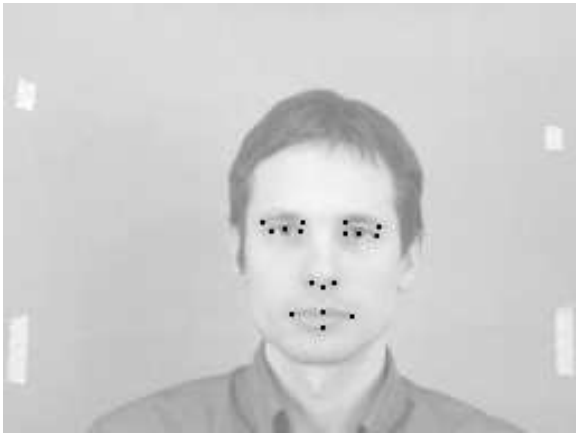

Figure 1.2: Some landmarks of the face.

landmarks which have been located manually. These include the corners of the eyes and the mouth, the midpoint of the top and the bottom edges of the lips or significant points of the nostrils and eyebrows. The image was made lighter so the landmarks can be easily seen.

The team of genetics researchers uses two approaches to locate forty landmarks in each face. One possibility is the manual selection, which can be in spite of its accuracy criticized as subjective and not scientific. When the landmarks are located repeatedly even by the same person, the results can be different. As the second approach the institute uses an automatic method, namely a software implemented by a commercial company, cooperating with the Institute of Neuroinformatics of the Ruhr University in Bochum. This software based on the algorithm Würtz (1997) and partially also on Wiskott et al. (1997) will be now described.

The algorithm starts by manual location of the set of 40 landmarks in a training set of 83 images of faces. These landmarks are called fiducial points and they together are placed on all positions in the image as one large template retaining fixed distances between the landmarks. Two-dimensional Gabor wavelet transformations with different values of the two-dimensional scale parameter are applied on all the training images and also on a new image in which the landmarks are to be located. The *jets* (Gabor wavelet coefficients) in each landmark of the training image and the jets in the corresponding pixels of the new image are compared. We can understand the jet of each of the training images as a (multi-dimensional) template. The correlation coefficient between the vectors of wavelet coefficients (or only their magnitudes) is computed and their sum over all 40 landmarks is used as the similarity measure between the training image and the new image.

The algorithm makes however more than just a comparison of the jets of the new image with the jets of each of the training images separately. It combines the jets of the new image with *any* combination of jets from different training images. Then the best similarity can be obtained with the mouth from one person, nostrils from another person and so on. Once these local experts are selected, some transformations are possible, for example local shift of particular points or scale transformations.

The everyday experience of the genetics researchers with the available software is however unsatisfactory because of its extremely high sensitivity to small rotations of the face. Actually Figure 1.1 is an example of a face in which the subroutine does not locate the eyes only because the eye on the right (the left eye of the person) is lower than the other one although this difference

Figure 1.3: An image with the face rotated by +45 degrees.

is very small, namely two pixels.

In this thesis we work with a database of grey-scale pictures of different people, which have been taken at the Institute of Human Genetics. The images are not accessible for public. Therefore we show the image of dr. Stefan Böhringer (Figure 1.1), the leader of the genetic research until Autumn 2005.

Each picture is a matrix with the size 192×256 pixels. A grey value in the interval $[0, 1]$ corresponds to each pixel, where low values are black and large values white. Pictures are taken under the same conditions, with the person sitting straight against the camera looking straight at it. The Institute tried to have the images standardized as much as possible. For example there are no images with closed eyes, hair over the face covering the eyes or other nuisance effects. Still the faces in the images happen to be rotated by a small angle. The eyes are not in a perfectly horizontal position in such images. The database does not include images with a three-dimensional rotation (a different pose).

While the whole database contains 212 images, we have divided it into two parts with 124 and 88 images, respectively. If not stated otherwise, we work with the 124 images of the first database. Only to check the performance of the methods we sometimes use the remaining 88 images and we refer to them as the new database.

The aim of this work is to search for the mouth and eyes in images of faces using templates. From the practical point of view it is actually desirable to search for landmarks rather than the mouth and eyes, but a natural first step is to find the mouth and eyes themselves. The information about their position in a given image simplifies the future task of locating the landmarks, which are prominent points (not only) of the mouth and eyes.

The practical performance of the methods is mostly desirable. Therefore we apply the methods not only to standardized images of the given database, but also to images with a different size or rotation of faces. In the text we consider only the rotation in a plane with the whole face well visible from the front. An example is shown in Figure 1.3 with the face rotated by +45 degrees. Another important aspect of the methods for locating landmarks in faces is robustness to noise in images.

Images of faces are very complex and we do not restrict ourselves to parametric models. A general criticism of likelihood approaches can be found for example in Davies and

Gather (2004) and is urgent so much the more in the image analysis: we believe that neither likelihood nor any other single-valued description can capture the complexity of highly dimensional data. At the same time this is a criticism of many of the existing approaches to image analysis, which are mentioned in Chapter 1.2 and to which we present an alternative.

This thesis is organized as follows. Chapter 1.2 presents a survey of relevant references for image analysis of faces. Chapter 2 is devoted purely to templates. We describe statistical measures of similarity between the template and the image and pay a special attention to the weighted correlation coefficient. Our construction of templates for locating the mouth and eyes is there explained. The performance of different templates is presented and compared.

Chapter 3 devoted to optimization describes the most important contribution of this work. The weighted correlation coefficient is used as the similarity measure between the template and the image and the weights are optimized to increase the separation between those parts of the face which correspond to the template and those which do not. Optimization of the weights without constraints tends to degenerate and to obtain a robust version we bound the influence of single pixels. Two algorithms are applied to find a reasonable approximation to the solution of this highly nonlinear optimization problem in a high dimension. This is a general method for optimization templates, not using any special properties of mouths. It is applied in Chapter 3.9 to optimizing the weights for eye templates. To improve the separation further the mouth template is optimized in Chapter 3.10 while the optimal weights are retained. Better results are obtained by optimizing the template at first and a consequent optimizing the weights for this template. The idea of optimizing the template can be understood as robust nonparametric discrimination.

Finally Chapter 4 presents some other results in locating the mouth, eyes and nose. There we search jointly for the mouth and eyes using templates rotated by all possible angles. Such approach can be used to find the rotation of the face and turns out to be a rotation-invariant procedure. The face can be rotated to the upright position with eyes in the horizontal position. Then the nose is simply located in a rectangle between the mouth and the eyes. Although we do not search for other landmarks of the face, we believe that such search would be relatively less complicated when the information about the position of the mouth and eyes is already known. The available commercial software can be also applied to search for other facial features. Thus the disadvantage of the existing software of being sensitive to rotation is removed.

There is a CD enclosed to the thesis containing source codes programmed in C and R. These programs are a supplement to Chapter 3 on optimization of the weights or templates over the whole database of images and there are several other files with auxiliary programs described in the file `"readme.txt"`, which is also present on the disc.

I would like to thank Professor P.L. Davies for introducing me to the topic of image analysis, for being the advisor of this thesis and throughout the whole period of my employment at the University of Duisburg-Essen and for many valuable ideas and suggestions.

I appreciate the continual financial support of the grant SFB 475 (Reduction of Complexity for Multivariate Data Structures) of the German Research Council (Deutsche Forschungsgemeinschaft, DFG). This work has been part of its section A1 (Robust Models and Dimension Reduction). I am thankful to the Institute of Human Genetics of the University Clinic Essen for the access to their database of images. These were taken as a part of the grants BO 1955/2-1 and WU 314/2-1 of the DFG.

1.2 Existing Methods

This chapter presents a survey of some references which are relevant for image analysis of human faces. Most of the existing work in the image analysis has been performed on grey scale images and this chapter also considers typically grey scale images of faces. While we focus on methods in the context of faces themselves, we also include remarkable and inspirative methods applied to images of other objects. In general, image analysis combines methods of mathematics, statistics and informatics as well as heuristic ideas which are tailor-made to suit the particular data and the particular task.

We have observed some general structure which is common to many of the methods for image analysis of faces. In very different methods for locating faces or recognizing the persons we can distinguish three typical procedures which are often carried out in the following order:

- Reduction of dimension
- Feature extraction
- Classification

The methods for the reduction of dimension include principal components, discrete Fourier transform, discrete cosine transform, wavelet transform and many others. The goal of feature extraction is to describe the differences among groups, to reveal the dimensionality of the separation among groups and the contribution of variables to the separation (Rencher 1998). This is often performed by the Fisher's linear discrimination. A classification procedure is often applied on the data obtained after the two previous transformations. This can be in principle any classification method, such as neural networks, nearest neighbours, logistic classification and so on.

Yang et al. (2002) give a survey of about 180 recent articles for the field of face detection and face recognition. As the main methods for locating faces the paper mentions neural networks, support vector machines, template matching, and hidden Markov models. Such survey is still not exhausting and contains details only of selected remarkable specific approaches. This chapter however aims at a general description of different methodologies, including more recent references and also inspirative ideas from other fields of image analysis.

We are aware of the fact that a systematic classification of references to different disjoint categories is problematic. We divide Chapter 1.2 to paragraphs, although particular practical approaches usually combine more of the different methods or tools described below.

We start with a paragraph on template matching, which is the topic of this thesis. Subsequent paragraphs are devoted to multivariate and Bayesian statistical methods, neural networks, support vector machines, classification trees, shape analysis and multiscale or multiresolution approaches. The final topic is three-dimensional image analysis of faces.

Template matching

Template matching is a tailor made method for object detection in grey scale images which uses the information about the ideal shape. Yang et al. (2002) describe it as one of standard methods for locating faces or their landmarks in a single image. They categorize and evaluate various approaches such as templates for the whole face; subtemplates for the eyes, nose or mouth; templates for face silhouttes; combinations of these approaches; or deformable templates where

one searches for the parameters giving the best fit in edges, peaks and valleys. We have not found references on using templates in colour images.

Jain (1989) and James (1987) describe template matching as a method for detection of presence of objects in given scenes or detection of movement of objects in time. Therefore it has numerous applications for example in weather prediction from satellite images, diagnosis of diseases from medical images, automation using robot vision and others. However the books do not discuss methods for construction of templates nor their optimization.

Dobeš et al. (2004) work with a training set of eye irises of 64 persons, which contains three images of the left iris and three images of the right iris of each person. Further they have a database of other iris images of the same persons who are in the training set and the aim is to assign every of the new iris images to the correct person. The pictures are taken as colour ones as usual RGB images, containing the red, green and blue layer. For the image analysis only red components of the images are used. The method uses a sophisticated algorithm to approximate the mutual information between a template and the image. The decision whether two iris images belong to the same eye is based on comparing the mutual information with a certain threshold.

Amit et al. (1991) consider hands as continuous mappings of a hand template degraded by noise. As a prior knowledge, the noise is assumed to have Gaussian distribution and the mapping is described by a two-dimensional Gaussian random field. The aim is to restore the image, in other words to remove noise to get the real hand which will be purely a Gaussian deformation of the template. The restoration is performed by sampling from the posterior distribution using MCMC approach.

Grenander (1993) creates representations of patterns in terms of algebraic systems and analyzes the structures from perspectives of algebra, topology, probability theory and statistics. These are applied to locating landmarks in leaves of a certain tree species or hands of different people. The book considers every image to be a deformation of a theoretical, pure image (template). For example every hand I^D is a deformation of a hand template I. The density $f(I^D|I)$ describes the probability distribution of all hands or in other words the distribution of the deformations and can be estimated from the data. The prior density is assumed to be Gaussian. Such image I^D is classified to be the hand which maximizes the posterior density $f(I|I^D)$ given by the rule of Bayes. In contrast with templates such approach combines ideal shape together with individual variability of smaller features, but on the other hand the approach is limited by its assumptions.

Downie et al. (1996) propose a method for deforming the template to make its shape more similar to the image. The deformation is a stochastic function modelled using a wavelet transformation, where each of the wavelet coefficients is a Gaussian random variable. This deformation function is selected from a class of functions, which is be sophisticated enough to allow for different deformations but at the same time contains only smooth and simple functions. A loss function combining good fit for the data with the smoothness and simplicity is minimized, which is equivalent to penalized least squares.

This method is applied by Downie and Silverman (2001) to two-dimensional (2D) images of archaeological skeletal remains. The difference of each image from the template is caused not only by the natural variation between biological features, but the bones can be also deformed in shape by osteoarthritis or completely broken. The value of the loss function then allows to classify the bones to three groups: bones with polished areas of the surface; bones damaged post-mortem; and bones without any of these traits. Ramsay and Silverman (2002) located a set of 12 landmarks manually in each of these images. After applying the Procrustes transform they computed principal components from the coordinates of the landmarks to study principal shapes

of the bones.

Yuille et al. (1992) use a parametric description for templates in grey scale images of faces. The eye template consists of two pieces of parabolas corresponding to the top and bottom contour of the eye. Further the template contains a circle in the middle corresponding to the pupil. Such templates are placed on every position in the image and parameters of the curves giving the best fit are found. Adjusting the parameters corresponds to altering the rotation, size and shape properties of the template. The position with the best fit over the whole image is then classified as the eye. Similarly a mouth template is described by two parabolas for the upper contour of the upper lip and another parabola for the bottom contour of the bottom lip. Such method of locating landmarks is a method tailor-made for the mouth and eyes and requires the perfect knowledge of the shape of these landmarks.

Multivariate statistical methods

Principal components analysis and the Fisher's linear discrimination are probably the most popular multivariate statistical methods applied in the image analysis. These are described in standard textbooks, for example Rencher (1998).

Hancock (2000) uses principal components of faces to propose a system, which automatically generates new faces. He uses the word eigenfaces for eigenvectors computed from faces. He computed eigenfaces from a database of 20 grey scale images of faces avoiding all with longer hair. To generate a new face, eigenfaces are combined with random proportions and then random morphing according to the principal components of shape vectors (so-called eigenshapes) is performed. An animation of the effect of the first eigenshapes in different faces is accessible on the internet (Hancock 2005).

Performing the principal components analysis on unadjusted images is however problematic. Hancock (2000) observed the eigenfaces to be blurred, the method is too sensitive to the position of landmarks, so he first locates the landmarks manually, adjusts the scale and deforms the face so that it has the average shape and the landmarks have the average position.

We have the impression that researchers and practitioners sometimes apply statistical methods to the image analysis without effort to understand the results in details and interpret them. Nevertheless the following example shows that a simple interpretation is often inaccessible, when several transformations are applied in successive steps.

Belhumeur et al. (1997) work with a training set of grey scale images of five people, containing 66 images of each person. The aim is the face recognition of a new image of one of the five given persons. To recognize to which person from a database the new image corresponds, the method starts off by computing principal components. Each of the images is replaced by the vector of several most important eigenvalues. These are linear combinations of the given data and the corresponding coefficients are given by eigenfaces. Only first thirteen principal components are considered, which correspond to the largest portion of explained variability. This reduces the dimension dramatically. The very first three of them are however omitted, because they can be explained only by different lighting conditions. Then the Fisher's linear discrimination is applied. That replaces the eigenvalues by their linear combinations, which are given by the discriminant functions. The final classification uses only five such real values for each image and applies a nearest neighbour classifier on them.

Other experiments of Belhumeur et al. (1997) include an attempt to classify images to two groups: faces with glasses and without them. Here the same steps are used as in the previous paragraph.

Er et al. (2005) use a training set of 40 grey scale images for each of 10 persons, altogether 400 images. The aim is again the face recognition. Discrete cosine transform is applied to reduce the dimension, only 60 values after the convolution are used. In the next step the cluster analysis is applied on these values for each class (person) separately. The 10 classes are thus divided to subclasses which reduces their variability and also allows to handle nonlinear variations in each class. Further the Fisher's linear discrimination finds linear combinations of the data which contribute at most to the linear discrimination among subclasses. The vector of 30 values corresponds now to each of the original images, which reduces further the dimension of the data. Finally a radial basis function neural network is used to classify the data to subclasses, which gives the decision about assigning a new image to one of the 10 people.

We would like to mention that Er et al. (2005) is an example of many papers in which instead of the whole faces their standardized versions are examined. In this case only the inner parts of faces are used, without hair and without background. Moreover the sizes of faces are modified to bring the distances between landmarks to standard values. Another example is Hancock (2000) who deforms the faces to an average shape before computing the principal components. Although such transformations simplify the task significantly, they are not described in the paper in details. They themselves can be responsible for good performance of the face recognition methods. Therefore we do not focus on the performance of different methods in this chapter. The performance depends on many factors such as complexity of images and scenes, number of faces in images, differences in the size, rotation, facial expressions and so on and the results from different references are not directly comparable.

Bayesian statistical methods

Winkler (1995) approaches the image analysis from the Bayesian point of view. The primary concern of the book is the Bayesian point estimation, which typically demands to compute the expectation of the posterior distribution. This is often complicated and requires approximative algorithms such as simulated annealing. In a general setting, which can be applied to images, the book develops the theory of Markov chains on finite spaces, Monte Carlo Markov Chains algorithms and proves ergodicity results.

Further Winkler (1995) applies the Bayesian approach to texture analysis, which studies repetitive patterns similar to the texture of cloth, lawn, sand or wood. The books also suggests penalized least squares for denoising (robustifying) images, combining the requirement of smoothness of the image with the assumption of Gaussian white noise. Finally the book studies maximum likelihood estimation for Markov random field models.

Bayesian approach is used also in some other references in the next paragraphs.

Neural networks

A neural network is a form of a multiprocessor computer system with simple processing elements (artificial neurons), which are highly interconnected (Smith 2001). In pattern recognition applications, neural networks are trained to associate output patterns with input patterns. Then that output is associated with the input which is least different from its pattern (Stergiou and Siganos 1996). Neural networks are nonlinear statistical models. Their training is a search for such values of the parameters which make the model fit the data well. The initial values are chosen to make the model nearly linear and become more nonlinear as the training iterates (Hastie et al. 2001). Neural networks are typically overparametrized, so it is recommended to stop the optimization before reaching the global optimum. Moreover the optimization is often

unstable. Vapnik (1995) explains that neural networks are not optimal classifiers. They are however often combined with good heuristics, which explains their success.

Rowley et al. (1998*a*) use neural networks to find one or more faces in a given grey scale image. The system firstly learns examples of eyes, noses and mouths from a database of 1050 face images in which the landmarks have been located manually. Also about 8000 nonface images are used in the training procedure. This template-based approach learns the appearance typical for faces but rare for nonfaces. Then small rectangular parts of new images are examined and the landmarks (eyes, nose, mouth) are searched for. To reduce the number of false positive cases, they examine the neighbourhood of suspicious areas and also repeatedly apply the networks with random initial weights and with different choices of other initial conditions.

Rowley et al. (1998*b*) then further improve the method for rotated faces. Firstly a separate neural network (router) is used for a fast preprocessing of each rectangular window of the image. Each window is examined in 36 situations, where the i-th case represents the rotation of $i \cdot 10$ degrees. The router assumes that the window contains a face, tries to estimate its rotation and rotates it to the anticipated upright position. Then the previous approach of Rowley et al. (1998*a*) is applied to decide if the window belongs to a face or nonface.

Brunelli and Poggio (1992) state that neural networks can be understood as a regularization approach to approximation of multivariate functions. The paper uses neural networks to classify grey scale images according to gender. The classifier is trained in a database with multiple images of 20 men and 20 women. The eyebrows thickness, distance of eyebrows from pupils and the nose width turn out to be most discriminating geometrical features for the gender recognition. The paper describes a prototype of a male and female face.

Support vector machines

Vapnik (1995) proposes support vector machines as a new way to train classifiers, based on theoretical results involving the Vapnik-Chervonenkis dimension of a set of functions, Vapnik-Chervonenkis entropy and other theoretical concepts. Usual techniques to train classifiers are based on the idea of the empirical risk minimization (ERM), which corresponds to the asymptotic approach in the statistical context. On the other hand support vector machines operate on another induction principle, called structural risk minimization (SRM). This is a trade-off between two factors: the value of the empirical risk and the upper bound for a confidence interval, corresponding to the generalization error.

For linear classifiers the method searches for hyperplanes maximizing the margin between classes and minimizing a quantity proportional to the number of misclassification errors. This is equivalent to solving a quadratic problem with linear constraints. For nonlinear decision surfaces the data are often projected to a space of a higher dimension; the linear classification problem is solved there and linear boundaries in the enlarged space correspond to nonlinear boundaries in the original space.

Training support vector machines involves to find a suitable value of a tuning parameter, which defines the width of the margin between classes and thus makes the decision boundaries either smoother or on the other hand wigglier. This selection can be performed by cross-validation.

Osuna et al. (1997) presents an algorithm allowing to train support vector machines over very large data sets and applies it to face detection in grey scale images using a database of 50 000 training objects (faces and nonfaces). From these the algorithm learned 2500 support vectors, which are faces with the largest similarity to nonfaces and also nonfaces very similar

to faces. The classifier is based only on these support vectors ignoring all other data. The support vectors can be understood as automatically learned templates for objects that lie at the boundary between the two classes.

Classification trees

A good introductory material on classification trees is given by Hastie et al. (2001). The trees are conceptually simple, powerful, easy to be interpreted and able to handle missing values in an elegant way. A disadvantage is however their instability. A small change in the data can result in a very different classification procedure, which is a consequence of the hierarchical nature of constructing the tree.

Growing the tree corresponds to partitioning the feature space into strata. Usually in each node of the tree only one variable (splitting variable) is considered at the time; splitting according to a linear combination of several variables would destroy the interpretability. In each node the value of the splitting variable is compared with a certain threshold (split point).

A popular method for automatic growing of trees is called CART (Classification and Regression Trees). One split is optimized at the time and the splitting variable and split point are found as minimizers of the loss function which describes the overall missclassification error or overall entropy E of the tree. Very large trees use too specific properties of the data. A possible solution avoiding an overfit is to stop growing the tree when a certain (rather large) number of nodes is attained and the tree is then pruned. Then instead of minimizing E the cost complexity measure $E + \alpha|T|$ is minimized, where $|T|$ denotes the size of the tree and the additional parameter α is chosen for example using cross-validation or Akaike's information criterion.

Wu and Trivedi (2004) use a binary tree to model the statistical structure of the eyes. Their database contains 317 grey scale images. The classification tree is built in top-down fashion, separating conditionally independent features into different subtrees, while keeping more dependent features in the same subtrees. The features are simply pixel intensities in different parts of the image. The probabilities for the model are learned in a nonparametric way and the rule of Bayes gives the decision if the object is classified as eye or noneye. The paper does not interpret features of eyes important for this discrimination.

Shape analysis

Graf et al. (1996) work with a database of 40 colour images of different people. They perform the shape analysis consisting of several filters detecting certain shapes. They use morphological operations usually in the form of convolutions with certain kernels which enhance areas with a given shape. For example small ellipses are searched for and areas with large values after the convolution are taken to be suspicious eyes. Based on the suspicious areas for the eyes there are already suspicious regions where the mouth can be expected to be situated. These regions are examined in the next step. The suspicious mouths are then regions which have a lower grey values in the middle than in the neighbourhood.

An examination of the sizes, distances and orientations of suspicious eyes and mouths suggests several possible positions of the face. Finally a large oval template is applied on this suspicious areas to locate the whole face and thus the position of the mouth and eyes is confirmed. Further the paper examines colour and motion in multiple images of a video sequence to locate faces.

Multiscale or multiresolution approaches

One example of a multiresolution approach is the algorithm of Würtz (1997), which has been described in Chapter 1.1.

From other references in this field we would like to mention Starck et al. (1998), who give a variety of applications of the wavelet approach to image processing. With the help of two-dimensional wavelets they denoise or compress images, deform templates to find objects in a complex scene or describe motion of objects in a series of images. Again the wavelet transform is performed more times, namely with different scales.

Kanters et al. (2005) propose a top point representation for images. They convolute the images with Gaussian kernels and their first two derivatives. That corresponds to blurring the images with different resolution. Objects (landmarks in a generic sense) remaining after more different convolutions are called top points. Object retrieval using templates is then based only on the top points, which are compared with top points from the template. No global measure is used to measure similarity between the template and the image, which allows the method to find rotated objects or even incomplete pieces of the objects.

Lindeberg (1994) studies theoretical aspects of multiscale approaches. He gives the following example to explain the concept of resolution of visual perception. We can see the forest, the trees or the leaves, all by looking at the same object. Human vision finds the suitable scale intuitively. The book believes that this would be too complicated for a computer system and because no scale can be a priori preferred, a multiscale approach is recommended. In examples the book convolves images with Gaussian kernels with different values of the parameters.

Three-dimensional (3D) image analysis of faces

Huang et al. (2002) create a 3D face model from two pictures of each person, namely one from the front and the other rotated in depth (with a different pose). They use support vector machines to locate 14 landmarks (eyes, nose, parts of the mouth) independently on each other. A geometrical classifier then examines all configurations of suspicious areas to find the whole face. This is an example of a bottom-up approach, creating the face from its components and examining their mutual positions. Such approach turns out to be more reliable than up-bottom approaches, locating the whole face at once.

Hammond et al. (2004) work with three-dimensional captures of faces obtained with a photogrammetric scanner with multiple cameras. In the database of 430 images there are patients with two different syndromes deforming the face and a control group. The aim is the classification of faces according to the genetic syndrome of the person. The novel approach of this paper is a 3D visualization of the face from the data and the representation of the face. The method starts with the manual location of 11 landmarks and use the (generalized) Procrustes algorithm to calculate the mean landmarks. The deformation of each face is measured which brings the landmarks into precise alignment with the mean landmarks.

Bronstein et al. (2005) compare 2D and 3D approaches to face recognition. They consider all 2D methods to be too sensitive to illumination, rotation and pose of the head, facial expressions and use of cosmetics. A 2D image contains namely a too limited information compared with 3D data. They create manually 3D models of faces from the training set. A face from a new person is then mapped onto the facial surface of another person. The paper shows a picture of Osama bin Laden as example; the distances between the landmarks belong to another person, but the resulting image still can be easily recognized as the world number one terrorist. This shows the weakness of human vision which is based only on a two-dimensional information, namely on

a projection of the face to a plane. Moreover anyone can draw another face on his own face and this deceives any existing 2D face recognition method.

Apart from such experiments Bronstein et al. (2005) use a 3D scanner to create a database of 3D images of 157 different faces. They apply multidimensional scaling to facial surfaces. This is a method detecting meaningful underlying dimensions explaining observed similarities or dissimilarities. The resulting face recognition is based on geometric invariants of faces and therefore robust to rotation and pose of the face and other effects.

There are not many references on image analysis of colour images. Such images are most often represented in the RGB colour space containing the red, green and blue layer. A colour image is a five-dimensional structure stored in the form of three matrices of intensities from the interval $[0, 1]$. Although some references on colour images are cited by Yang et al. (2002), it would be very difficult to generalize most of the methods of this chapter for colour images.

Chapter 2

Templates

This chapter starts with an overview of possible measures of similarity between the template and the image. A special attention is paid to the weighted correlation coefficient (Chapter 2.2). Chapter 2.3 describes our construction of templates for the mouth and eyes and presents their performance. Chapter 2.4 presents the performance of the weighted correlation with radial weights in an example and later in the whole database of images. Finally Chapter 2.5 examines some further aspects of locating the mouth using templates. The templates are applied on the database of 124 images coming from the Institute of Human Genetics as described in Chapter 1.1.

2.1 Measures of Fit

Template matching is described in Chapter 1.2 as a tailor made method for object detection in grey scale images. A template is a model, a typical form, an ideal object. It is placed on every possible position in the image and the similarity is measured between the template and each part of the image, namely the grey value of each pixel of the template is compared with the grey value of the corresponding pixel of the image. The literature mentions the sample correlation coefficient as the standard similarity measure in this context. Such area is considered to be suspicious which has the largest value of the sample correlation with the template.

Let us consider two data vectors

$$\boldsymbol{x} = (x_1, x_2, \ldots, x_n)^T \quad \text{and} \quad \boldsymbol{y} = (y_1, y_2, \ldots, y_n)^T. \tag{2.1}$$

The sample correlation coefficient between the vectors $\boldsymbol{x}$ and $\boldsymbol{y}$ is defined by

$$r(\boldsymbol{x},\ \boldsymbol{y}) = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}[(x_i - \bar{x})^2] \sum_{j=1}^{n}[(y_j - \bar{y})^2]}},$$

where $\bar{x}$ and $\bar{y}$ are arithmetic means of the data vectors $\boldsymbol{x}$ and $\boldsymbol{y}$ respectively. It is invariant with respect to regression and scale.

James (1987) considers only the correlation coefficient to be a suitable similarity measure between two images. He understands the correlation as a convolution between the template and the picture.

Jain (1989) suggests to use the sum of squares as a similarity measure between the template and the image. Denoting the data as vectors $\boldsymbol{x}$ and $\boldsymbol{y}$ like in (2.1), the suggested measure is the

sum

$$\sum_{i=1}^{n}(x_i - y_i)^2.$$

It does not however possess the invariance properties to regression and scale. The book further presents a method for a fast search of the best direction in which the template should be shifted in order to improve the correspondence between the template and the image.

A weighted analogy of the sample correlation coefficient is described later in Chapter 2.2. Other possible measures of correlation include nonparametric and robust ones. We have not found any of these in the image analysis literature in the context of template matching. One of such possible measures of correlation is Spearmans's rank correlation coefficient.

Other possibilities are defined in the linear regression context. Then it is reasonable to transform both the template and the part of the image to vectors and perform linear regression of the part of the image against the template. We recall that the coefficient of determination R^2 in this regression fulfills $r^2 = R^2$, in other words it is up to the sign equivalent to the sample correlation coefficient. Other possibilities such as the least squares estimate of the slope in the regression or the residual sum of squares in the regression do not perform well, evidently because they are not invariant to linear transformations of the data.

A survey of robust measures of correlation is given by Shevlyakov and Vilchevski (2001). These can be classified to several groups, namely direct robust counterparts of the sample correlation, then measures based on robust regression, measures based on robust estimation of the variance of principal variables and finally correlation measures computed after a preliminary rejection of ourliers from the data. One simple possibility of an approach belonging to the last group is to classify a certain (given) percentage of outliers in the data by least trimmed squares and compute the sample correlation from the remaining data. After a regression or scale transformation the same data points are classified to be outliers thanks to the invariance properties of the least trimmed squares. Such approach is therefore invariant to regression and scale.

We can understand the task to find the landmarks (mouth, eyes) as nonparametric regression, where the model for the landmark is given by the template. There the grey value in each pixel is a parameter of the model.

2.2 Weighted Correlation Coefficient

We have not found any references on using the weighted correlation in the template matching context as a similarity measure between the template and the picture. We now give the definition and summarize basic properties of the weighted correlation, which are useful for later chapters.

The (sample) weighted correlation coefficient is defined for data vectors (2.1) and nonnegative weights $\boldsymbol{w} = (w_1, w_2, \ldots, w_n)^T$ by the formula

$$r_W(\boldsymbol{x},\ \boldsymbol{y};\ \boldsymbol{w}) = \frac{\sum_{i=1}^{n} w_i(x_i - \bar{x}_W)(y_i - \bar{y}_W)}{\sqrt{\sum_{i=1}^{n}[w_i(x_i - \bar{x}_W)^2]\sum_{j=1}^{n}[w_j(y_j - \bar{y}_W)^2]}}, \tag{2.2}$$

where

$$\bar{x}_W = \sum_{i=1}^{n} w_i x_i \Big/ \sum_{i=1}^{n} w_i \quad \text{and} \quad \bar{y}_W = \sum_{i=1}^{n} w_i y_i \Big/ \sum_{i=1}^{n} w_i$$

are the weighted means.

In the whole work we assume

$$\sum_{i=1}^{n} w_i = 1, \tag{2.3}$$

which is a standard assumption in the context of weighting. However formula (2.2) does not require this assumption and transforming the weights to have their sum different from 1 does not change the resulting value of r_W.

For computational purposes it is more convenient to use the formula

$$r_W(\boldsymbol{x}, \boldsymbol{y};\ \boldsymbol{w}) = \frac{\sum_{i=1}^n w_i x_i y_i - \bar{x}_W \bar{y}_W}{\sqrt{(\sum_{i=1}^n w_i x_i^2 - \bar{x}_W^2)(\sum_{j=1}^n w_j y_j^2 - \bar{y}_W^2)}}.$$

We stress that this is equivalent with formula (2.2) only under the assumption (2.3).

In spite of its simplicity we have not found the weighted correlation r_W itself in statistical literature. In the software package R there is the function `corr` computing the formula (2.2) only in the library `boot`, which is a special package for bootstrap. Now we describe some properties of the weighted correlation coefficient and then show that it is (up to the sign) equivalent to the well-known weighted coefficient of determination from the weighted regression context.

The weighted correlation clearly fulfills $r_W \in [-1, 1]$ for any data vectors (2.1) and any weights. If there is a perfect linear relationship between the two variables, then the weighted correlation attains 1 or -1, namely

$$r_W(\boldsymbol{x}, a\boldsymbol{x} + b;\ \boldsymbol{w}) = \mathsf{sign}(a) \quad \text{for any} \quad a \neq 0,\ b \in \mathbb{R}$$

and for any data vector $\boldsymbol{x} = (x_1, \ldots, x_n)^T$ and weights $\boldsymbol{w}$, where sign denotes the sign function.

The weighted correlation coefficient is invariant to a linear transformation of the data vectors $\boldsymbol{x}$ and $\boldsymbol{y}$. Namely for any data vectors (2.1) and weights $\boldsymbol{w}$ it holds

$$r_W(a\boldsymbol{x} + b, c\boldsymbol{y} + d;\ \boldsymbol{w}) = r_W(\boldsymbol{x}, \boldsymbol{y};\ \boldsymbol{w}) \cdot \mathsf{sign}(a) \cdot \mathsf{sign}(c) \quad \text{for any} \quad a \neq 0,\ b \in \mathbb{R},\ c \neq 0,\ d \in \mathbb{R}.$$

Thus r_W is a natural generalization of the classical correlation coefficient.

To show the connection between the weighted correlation coefficient and weighted regression, let us consider the regression model

$$y_i = \beta_0 + \beta_1 x_i + e_i, \quad i = 1, \ldots, n, \tag{2.4}$$

where the i.i.d. errors $\boldsymbol{e} = (e_1, \ldots, e_n)^T$ fulfill

$$\mathsf{E}\boldsymbol{e} = \boldsymbol{0} \quad \text{and} \quad \mathsf{var}\,\boldsymbol{e} = \sigma^2 \boldsymbol{W}^{-1} = \sigma^2 \mathsf{diag}\left\{\frac{1}{w_1}, \ldots, \frac{1}{w_n}\right\}.$$

Positive weights are now required. The response is a random variable and capital letters $Y_1, \ldots, Y_n$ should be used, but let us rather use the same notation both in the correlation and regression context.

Let us rewrite the model in the matrix notation as $\boldsymbol{y} = \boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{e}$, use the (generalized) least squares estimator $\boldsymbol{b}_W = (\boldsymbol{X}^T\boldsymbol{W}\boldsymbol{X})^{-1}\boldsymbol{X}^T\boldsymbol{W}\boldsymbol{y}$ and denote fitted values by

$$\hat{\boldsymbol{y}}_W = (\hat{y}_{W1}, \ldots, \hat{y}_{Wn})^T = \boldsymbol{X}\boldsymbol{b}_W.$$

Let us use the notation $\mathbf{1} = (1, \ldots, 1)^T$ and let $\bar{y}_W$ be again the weighted mean.

Now we consider the residual and total sums of squares in the weighted regression model. Their decompositions

$$\mathsf{SSE}_W = (\boldsymbol{y} - \hat{\boldsymbol{y}}_W)^T \boldsymbol{W} (\boldsymbol{y} - \hat{\boldsymbol{y}}_W) = \boldsymbol{y}^T \boldsymbol{W} \boldsymbol{y} - \hat{\boldsymbol{y}}_W^T \boldsymbol{W} \hat{\boldsymbol{y}}_W$$

and

$$\mathsf{SST}_W = (\boldsymbol{y} - \bar{y}_W \mathbf{1})^T \boldsymbol{W} (\boldsymbol{y} - \bar{y}_W \mathbf{1}) = \boldsymbol{y}^T \boldsymbol{W} \boldsymbol{y} - \mathbf{1}^T \bar{y}_W \boldsymbol{W} \bar{y}_W \mathbf{1}$$

allow us to define and then express the regression sum of squares in the form

$$\mathsf{SSA}_W = \mathsf{SST}_W - \mathsf{SSE}_W = \hat{\boldsymbol{y}}_W^T \boldsymbol{W} \hat{\boldsymbol{y}}_W - \mathbf{1}^T \bar{y}_W \boldsymbol{W} \bar{y}_W \mathbf{1} =$$

$$= (\hat{\boldsymbol{y}}_W - \bar{y}_W \mathbf{1})^T \boldsymbol{W} (\hat{\boldsymbol{y}}_W - \bar{y}_W \mathbf{1}).$$

It is natural to define the weighted coefficient of determination in this context by

$$R_W^2 = \frac{\mathsf{SSA}_W}{\mathsf{SST}_W} = 1 - \frac{\mathsf{SSE}_W}{\mathsf{SST}_W} = \frac{\sum_{i=1}^n w_i (\hat{y}_{Wi} - \bar{y}_W)^2}{\sum_{i=1}^n w_i (y_i - \bar{y}_W)^2}.$$

Now it follows that $R_W^2 = r_W^2$. Later in the text we also need the following lemma.

Lemma: In the weighted regression model (2.4) with weights $\boldsymbol{w} = (w_1, \ldots, w_n)^T$, let the residuals by defined by

$$\boldsymbol{u} = (u_1, \ldots, u_n)^T = \boldsymbol{y} - \hat{\boldsymbol{y}}_W.$$

Then it holds

$$r_W(\boldsymbol{x},\ \boldsymbol{u};\ \boldsymbol{w}) = 0.$$

Proof: The numerator of the desired weighted correlation coefficient equals

$$\sum_{i=1}^n w_i x_i u_i - \left(\sum_{i=1}^n w_i x_i \right) \left(\sum_{j=1}^n w_j u_j \right).$$

The first sum is zero because of $\boldsymbol{X}^T \boldsymbol{W} \boldsymbol{u} = 0$. The weighted sum of residuals is also zero, which follows from the set of normal equations for the weighted regression model.

Finally let us remark that the weighted correlation is not equivalent to the sample correlation coefficient from the data vectors

$$\boldsymbol{x}^* = (\sqrt{w_1}\, x_1, \ldots, \sqrt{w_n}\, x_n)^T \quad \text{and} \quad \boldsymbol{y}^* = (\sqrt{w_1}\, y_1, \ldots, \sqrt{w_n}\, y_n)^T.$$

Such modification transforms the weighted regression to ordinary least squares, but such model does not contain an intercept and neither the coefficient of determination nor the correlation coefficient have a reasonable interpretation. There is actually no such transformation of the data so that the classical correlation coefficient of the transformed data would correspond to the weighted correlation of the original data.

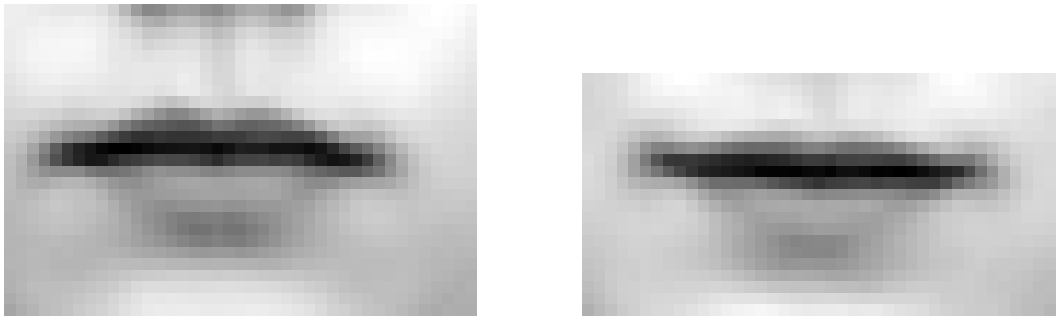

Figure 2.1: Mouth templates.

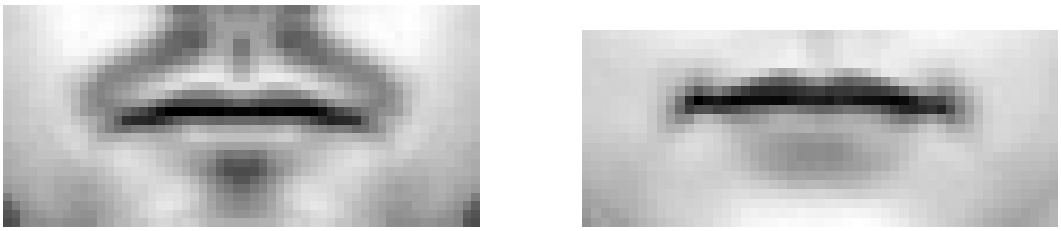

Figure 2.2: Mouth templates.

2.3 Construction of Templates

Templates for the Mouth

We describe our approach to the construction of templates for the mouth. Firstly we have considered a mouth from one of the pictures, which looks more or less typically. We take it as a rectangle of size 27×41 pixels containing the lips and their neighbourhood and place on every possible position in all the 124 pictures. We use the sample correlation to compare grey values from the template and every part of the image, after transforming them to vectors of length $27 \times 41 = 1107$ pixels.

In 16 pictures the maximal correlation between the template and the image exceeds 0.85 and this largest correlation is obtained each time in the mouth. The average of the grey values of the 16 mouths is shown in Figure 2.1 (left) and that is used as the first template. About 90 % of the 124 mouths have a larger correlation with this first template than with the initial mouth. The process of averaging removes individual characteristics and retains typical properties of objects.

The procedure was repeated always with such initial mouth, which did not have the correlation with any of the previous mouths above 0.80. Some mouths in the database are not standard, for example not horizontal, open with visible teeth, smiling or with light lips after using a lipstick. These were taken also as initial templates. Seven of them were not highly correlated with any of other mouths, so the averaging could not be performed.

In the whole database with 124 images there are five pictures with a beard and/or moustache, quite different from each other in terms of the size, position and grey values of the bearded areas. We created also two templates with a beard. One of them is shown in Figure 2.2 (left).

Altogether a set of 13 mouth templates of different sizes was created. Some of them are rather small rectangles including just the mouth itself and the nearest neighbourhood, others go as far downwards as to the chin. In general, when several templates are used at the same

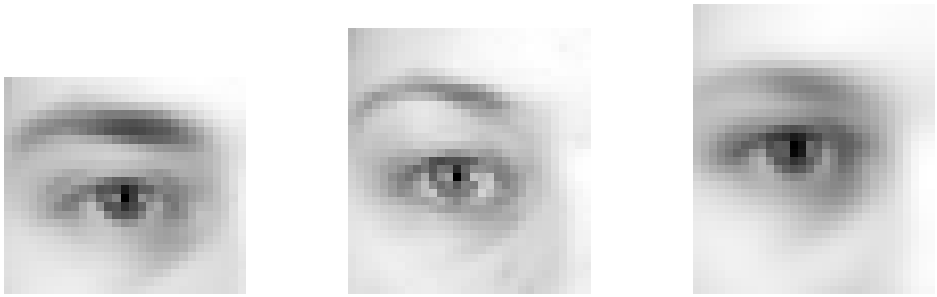

Figure 2.3: Some of the eye templates.

time, that rectangular area of the picture with the largest correlation with any of the mouth templates is classified as the output of the template matching. In other words we search for the maximum over the whole image of the maximal values over different choices of the template. All the 13 templates together lead to correct locating the mouth in every of 124 examined pictures, when the correlation coefficient is used as the measure of association between the template and every area of the picture.

It is possible to reduce the number of templates but then the correlation between the template and the mouth does not necessarily exceed the previous bound 0.80. We have started with one particular template and averaged such mouths, which have the correlation with it over 0.80. This mean represents one of the new templates. Then we selected another of the previous templates and performed the same procedure. The selection of templates from the set of thirteen templates was subjective and we have tried to select templates, which would be very different from those selected in previous steps. Finally when the number of newly created templates reached seven, it was possible to locate all the mouths and more templates were not needed. Then however the correlation between the template and the mouth drops as far downwards as to 0.60.

The final set includes seven templates with different sizes, namely two templates with a beard and five without it. Some of them are shown in Figures 2.1 and 2.2. The seven templates together locate the mouth correctly in every image in the database with 124 images. We must specify precisely in which situations is the mouth located correctly, in other words to find the boundary between mouths and nonmouths. In this chapter we decide only subjectively if the mouth is located correctly and namely we accept suspicious areas with the midpoint not more than a few pixels distant from a subjectively located midpoint of the mouth. Later in Chapter 3 we proceed more precisely in defining the mouth area.

Templates for the Eyes

We have created the templates for the right eyes in the same way as the mouth templates. These are right eyes of every person, so they are actually on the left side of the images. Some of the templates are shown in Figure 2.3. Their size equals 26×29 pixels, 33×30 and 36×30 pixels respectively. Altogether we have created six eye templates. The remaining ones are either squares or rectangles of similar sizes with the templates from Figure 2.3 and all of them contain eyebrows. We take the templates for the other eye as mirror reflections of these six templates and search for both eyes without any assumption on their distance or their expected position.

It happens however that for example the right eye of a particular person has a larger correlation with one of the left eye templates than with any of the right eye templates. We use this phenomenon in the algorithm in the following way. All the twelve templates are placed on every

Figure 2.4: Left: radial weights. Right: the same image in the log scale.

possible position in the image. We do not distinguish between left and right templates. First the area with the largest correlation over all templates is found. That is one suspicious eye. Now the whole image without this suspicious area and its nearest neighbourhood is considered. Again the area with the largest correlation with any of the eye templates is found. That is the other suspicious eye. The eyes are found correctly in 100 % of images.

To locate more precisely the midpoint of the pupil, we take the darkest pixel in the neighbourhood of the most correlated point. The neighbourhood is selected as a circle with radius five pixels. The results are acceptable in 99 % of images. The only one failure happens in the picture of one lady. The suspicious area moves to the eye lashes which are darker than the iris, seemingly because of cosmetics.

2.4 Weighted Correlation With Radial Weights

In this section we define the radial weights, explain the motivation for using them, show an example explaining the good performance of radial weights and apply them to locating the mouth in the whole database of 124 images of faces.

The weighted correlation coefficient with equal weights is equivalent to the sample correlation coefficient (without weights). In order to stress the importance of the lips or in general the central area of the template for locating the mouth it is natural to use the weighted correlation coefficient with radial weights. Then the weight of every pixel is inversely proportional to its distance from the midpoint. Figure 2.4 (left) shows radial weights of the size 26×56 pixels, which is the size of the bearded template (Figure 2.2 left). White pixels correspond to large weights there. Because only the large weights in the midpoint can be recognized in the image, we give a plot in the log scale in the right image of Figure 2.4. There it can be seen that weights decrease as the distance from the pixels increases. The radial weights are simple and at the same time reasonable, because the lips define the mouth certainly more than the corners of the template. However when several templates are applied at the same time and radial weights are used, then the weights correspond to the size of the template and are therefore different from each other.

For the radial weights of size 26×56, let us denote the coordinates of the (virtual) midpoint by $[m, n] = [13.5, 28.5]$. In a particular pixel with coordinates $[i, j]$ we determine

$$w_{ij}^* = \frac{1}{\sqrt{(i-m)^2 + (j-n)^2}} \tag{2.5}$$

and the weight in the pixel then equals to w_{ij}^* multiplied by a constant such that the sum of the weights equals 1. If the number of rows and/or columns is odd and (2.5) is not defined in the midpoint, then we assign the same weight to the midpoint as to its direct neighbours.

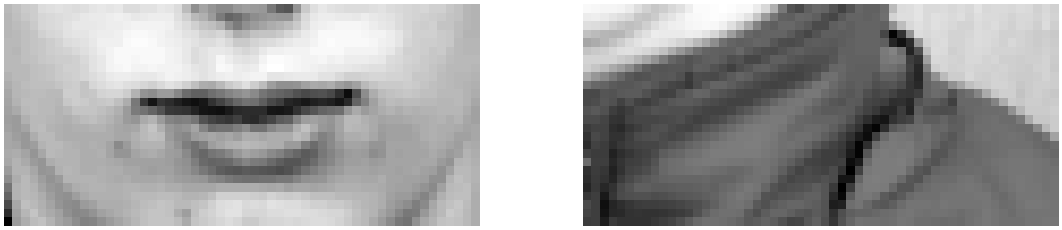

Figure 2.5: A mouth and a nonmouth.

Table 2.1: Sums of squares in the example of Chapter 2.4: regression of the mouth, resp. nonmouth against the bearded template.

Sum of squares	Mouth	Nonmouth
SSA	10.3	18.0
SSE	35.4	48.8
SST	45.8	66.8
R^2	0.23	0.27
r	0.48	0.51

Computing the weighted correlation between the template and the part of the image, both of these matrices and also the weights must be transformed to vectors. The seven templates together with the radial weights locate the mouth correctly in each of the 124 pictures. We remark that the templates were created in a way which depends on results obtained with the classical correlation, equivalent to weighted correlation with equal weights.

In locating the mouth the weighted correlation with radial weights brings improvements compared to the sample correlation except for some exceptions. One of them is a template which has the lips in its upper part. Then its center is a homogeneous area between the bottom lip and the chin. Another example is the template from right part of Figure 2.1. The bottom lip is contained in the middle, obtaining the largest weights. Its rounded shape causes areas with a similar shape (for example the chin) to have larger weighted correlation with the template and the chin turns out to be more suspicious than the mouth in some pictures, in which the classical correlation finds the mouth correctly. In general the radial weights bring improvements for templates, which have the midpoint of the lips precisely in the middle. The improvement in locating the mouth with radial weights will be even more appreciable later in images with a different size or rotation of the face.

To locate the mouth in one image using seven templates, our subroutine programmed in the language C needs 27 seconds. For comparison a subroutine in the software package R needs 84 seconds.

The weighted correlation coefficient with radial weights brings improvements also in locating the eyes. Both the right and left eyes are located correctly in all 124 images of the database. The performance in locating the eyes in faces with a different size or in faces rotated by a small degree is however poor.

Example—Effect of the Weighted Correlation

In this chapter we consider the two images from Figure 2.5. One of them is a mouth of one lady and the other is a part of her shirt. The aim is to classify the mouth correctly, in other words to discriminate between the mouth and the nonmouth using the bearded template from Figure 2.2. This example was selected because it shows the dramatic improvement of the weighted correlation with radial weights compared to the classical correlation.

In this example the sample correlation between the mouth and the bearded template attains only 0.48, while the sample correlation between the nonmouth and the template is larger, namely 0.52. Therefore the nonmouth is more suspicious to be the mouth than the true mouth itself. Although the bearded template is not the best possibility to search for a nonbearded mouth, a more detailed study shows that the main problem is using the sample correlation as the similarity measure.

The next explanation is based on Figure 2.6. The left plot contains grey values of the bearded template on the horizontal axis and the mouth from Figure 2.5 on the vertical axis. Actually both variables are standardized to have the values in the interval $[0, 1]$, but this influences neither the classical nor the weighted correlation coefficient between them. First of all the image shows that it is reasonable to use the linear model for the relationship between the mouth and the mouth template.

We recall the sample correlation coefficient is (up to the sign) equivalent to the coefficient of determination in the linear regression. This was explained in Chapter 1.2 for the weighted case. The classical sample correlation coefficient compares the variability explained by the linear model with the total variability in the linear regression.

In the left image of Figure 2.6 the red line corresponding to the least squares regression fit of the mouth against the template gives a reasonably good fit. The blue horizontal line corresponds to the mean of grey values of the mouth and represents its more or less typical value. The sample correlation is based on comparison of the linear model (red line) with the submodel (blue line).

In the right image of Figure 2.6 there are grey values of the nonmouth plotted against those of the template. The pixels in the top right corner corresponding to the top corners of the nonmouth have a strong leverage effect. Also the pixels of the lips in the left bottom corner of the scatter plot have a slight leverage effect. These two leverage effects cannot push the red line upwards because the linear regression line must go through the center of gravity of the data, which is the intersection point of the red and blue lines. The leverage effect therefore tries to rotate the red line, but the two effects work in contradictory directions. The blue line in the right image is not a suitable submodel. The mean is not a typical value of the response and for such mixture of two clusters a suitable submodel can be hardly found.

Now we can compare the two scatter plots of Figure 2.6. In the left image the correlation compares a good model with a relatively good submodel, while in the right image a weak model with a very poor submodel. That makes the correlation on the right larger than in the left image, although the linear model itself is poorer in the right image. This problematic behaviour is the fundamental property of the sample correlation coefficient.

A better insight to this example can be obtained in Table 2.1. The left column considers linear regression of the mouth from Figure 2.5 against the bearded template and gives numerical values of the sums of squares together with the coefficient of determination and the sample correlation between the mouth and the template. The right column contains analogous values for the nonmouth from Figure 2.5 compared with the same template. Using the notation of Chapter 1.2, SSE and SST are the error sum of squares and total sum of squares respectively.

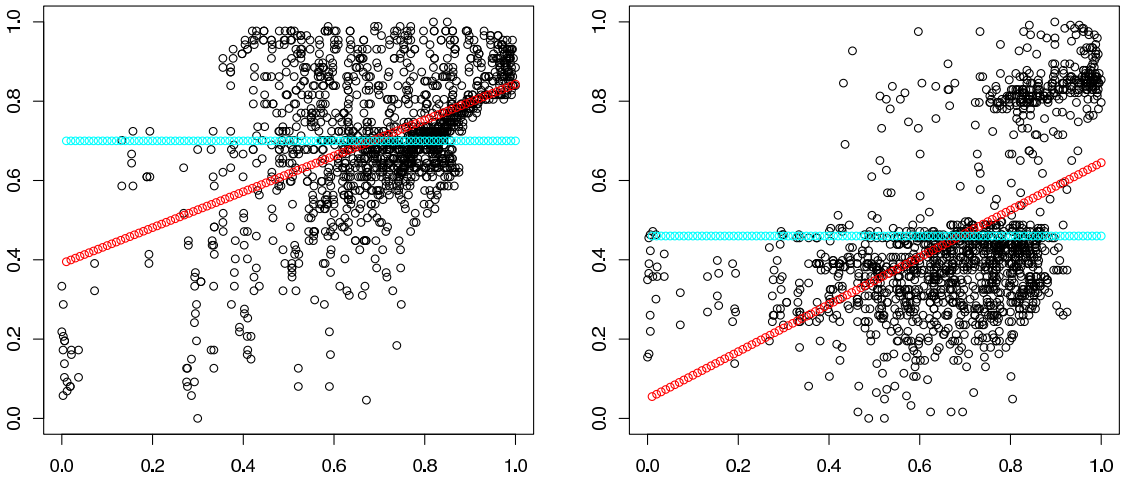

Figure 2.6: Left: mouth against the mouth template. Right: nonmouth against the same template. Least squares regression (red) and arithmetic mean (blue).

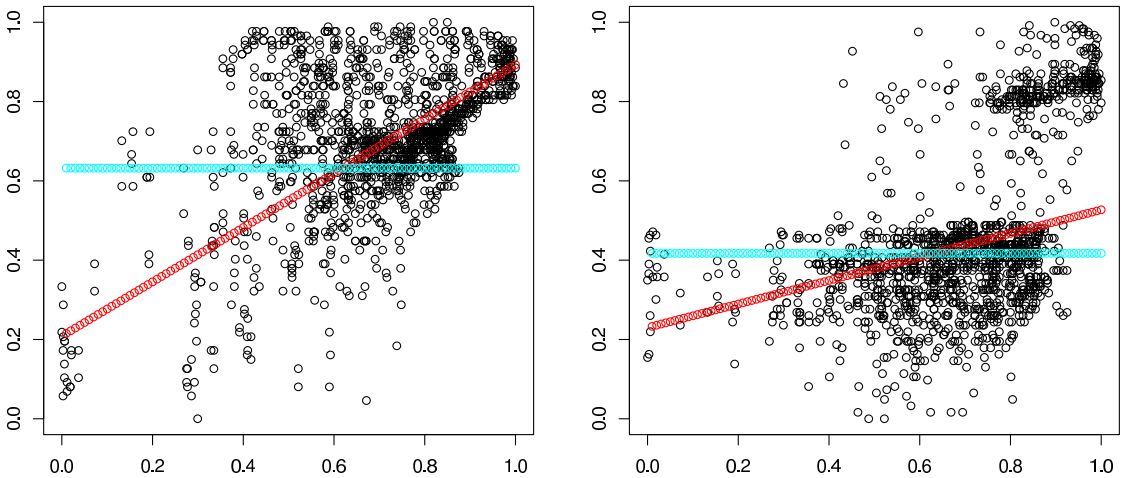

Figure 2.7: Left: mouth against the mouth template. Right: nonmouth against the same template. Weighted regression (red) and weighted mean (blue) with radial weights.

These are sums of squares in the situation without weighting (in other words with equal weights). The first row of Table 2.1 contains the regression sum of squares defined as $\mathsf{SSA} = \mathsf{SST} - \mathsf{SSE}$. In the last rows there is the coefficient of determination equal to the ratio $\mathsf{SSA}/\mathsf{SST}$ and this is at the same time the square of the sample correlation r. Based on Table 2.1 there is a larger variability of the response and also a larger regression sum of squares SSA in the right image of Figure 2.6 than in the left image. The weighted coefficient of determination is larger in the right image and this is based on comparing the quality of the linear model with the quality of the submodel, in other words the regression sum of squares is compared to the total sum of squares.

Next we examine the weighted correlation with radial weights. The weighted correlation between the mouth and the template equals 0.66 and between the nonmouth and the template 0.38.

We explain why does the weighted correlation outperform the classical correlation in locating the mouth. This is shown in Figure 2.7, where the red lines correspond to the weighted regression with radial weights and blue lines to the weighted means with the same weights. There is again the mouth plotted against the template on the left and the nonmouth against the template on the right and these can be compared with Figure 2.6.

In the regression of the mouth against the template the left bottom cluster of data corresponds to the lips which get larger weights. Therefore they become more important leverage points, the red line describes them better and that improves the weighted correlation. We can also see that the regression line becomes steeper.

In the right image of Figure 2.7 there is the nonmouth fitted against the template. The lips situated again in the left bottom corner of the scatter plot are now important leverage points, while the cluster in the top right part becomes less important and worsely explained by the red line. Moreover the red line becomes closer to the horizontal blue line. The weighted correlation then compares two models not so different from each other any more, the weighted correlation between the nonmouth and the template is therefore lower than the sample correlation and the part of the shirt becomes less suspicious than the true mouth.

To summarize, the choice of weights for the weighted correlation influences the steepness of the regression line, which further affects the value of the weighted correlation coefficient. A suitable choice of weights can improve the discrimination between a mouth and a nonmouth. In Chapter 3 we optimize the weights to improve this discrimination and to make the procedure of locating the mouth more reliable.

Locating the Mouth With One Template

We have performed the following experiment. First we have made all templates symmetric. Then we use each template separately to locate the mouth. The top part of Table 2.2 gives the percentage of cases with the mouth located correctly in the database with 124 images. The table gives the size of these templates and results for the classical sample correlation r and weighted correlation r_W with radial weights. The last measure is the Spearman's rank correlation coefficient r_S corresponding to the sample correlation computed with data which have been converted to ranks.

In the top part of Table 2.2 there are results of locating the mouth with just one template at the time. Five templates are nonbearded and two are bearded. Some of the templates are shown in Figures 2.1 and 2.2. The nonbearded templates in these images are averages of mouths of different people. The bearded template of Figure 2.2 is the average of four bearded mouths.

Table 2.2: Percentages of images with the correctly located mouth using different templates. Comparison of the sample correlation, weighted correlation with radial weights and Spearman's rank correlation.

Template with description	r	r_W	r_S	Size of the template
1. Nonbearded (Figure 2.1 left)	0.93	0.94	0.80	27×41
2. Nonbearded (Figure 2.1 right)	0.94	0.91	0.82	21×41
3. Nonbearded (Figure 2.2 right)	0.99	0.99	0.83	21×51
4. Nonbearded	0.92	0.69	0.83	21×41
5. Nonbearded	0.95	0.96	0.60	26×41
6. Bearded (Figure 2.2 left)	0.91	1.00	0.50	26×56
7. Bearded	0.62	0.78	0.43	29×56
Combinations of these templates:				
1 + 2 + 3 + 4 + 5 + 6 + 7	1.00	1.00	0.94	
1 + 6	0.98	0.97	0.82	
1 + residuals	0.95	0.98	-	
6 + residuals	0.98	0.98	-	
1 + residuals + 6	0.98	0.99	-	
Eigenmouth (Figure 2.9)	0.89	0.99	0.35	26×56
6 + eigenmouth	0.94	1.00	0.88	
1 + eigenmouth	0.96	0.94	0.79	

The other bearded template from Table 2.2 is a bearded mouth from only one image, which is not well correlated with other bearded mouths. Only later we comment using more templates at the same time and describe the bottom part of Table 2.2.

The nonbearded template from the fourth row of Table 2.2 contains an area below the mouth as far downwards as to the chin, so the lips are present at the top of the template. The lips thus get assigned smaller weights and our choice of weights is then not optimal. We have observed the weighted correlation to perform better than the sample correlation for the situation when the largest weights correspond to the center of the mouth.

It turns out that one of the templates gives correct results in 100 % of cases. This is namely the bearded template (Figure 2.2 left) used with radial weights. In general the template should fulfill two contradictory requirements to get as reliable results as possible: it should resemble a real mouth, but on the other hand should not resemble any nonmouth. In this case only the template with a beard turns out to be strong in both properties.

The Spearman's rank correlation has a low performance. It is at the same time computationally intensive. It takes 80 seconds in the package R to compute the rank correlation between one template and every part of one particular image. Our subroutine programmed in C using the quicksort algorithm reduces the speed to 19 seconds. Results of locating the mouth using the rank correlation are included again in Table 2.2.

In the bottom part of Table 2.2 there are results of using several templates together. There we compute the correlation between the image and each of the templates separately and then take the maximum. The size of the templates is not repeated there again. The residuals and eigenmouth will be explained in later chapters.

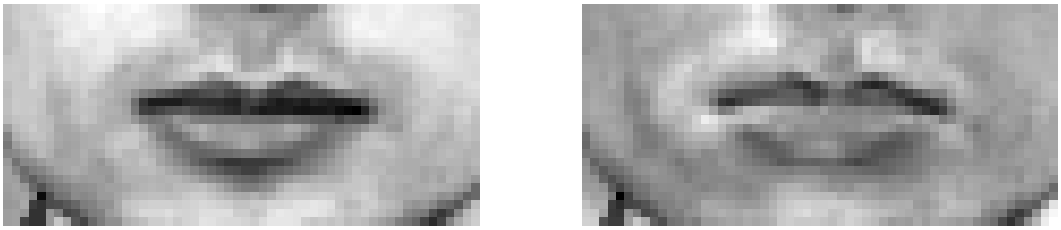

Figure 2.8: Left: the mouth of the person from Figure 1.1. Right: residuals of the linear regression of that mouth against the bearded template from Figure 2.2.

A combination of two templates, namely the bearded and nonbearded ones from the first and sixth rows of Table 2.2, has a worse performance in comparison with the bearded template itself, because some of nonbearded pictures become problematic. The bearded template locates correctly the mouth even in such images of nonbearded people, for whom the nonbearded template fails. However the nonbearded template is highly correlated with such areas as the chin, hair, eyebrows or others and this correlation happens to be larger than the weighted correlation between the bearded template and the mouth. Using both templates together inherits properties of the weaker of the template.

We have also examined what happens with weights equal to the template. We take the bearded template from Figure 2.2 (left). The same image transformed to have the sum of the grey values equal to 1 is taken as the matrix of weights. The mouth is in this located correctly only in 86 % of images. There is no reason for the weights to copy exactly the structure of the template.

2.5 Locating the Mouth Using Templates

In this section we describe some further results in locating the mouth. We examine the residuals of linear regression of a mouth against the template, compute principal components analysis of mouths and finally examine the robustness of locating the mouth with templates to a different size or rotation of faces.

Residuals

Now we examine the residuals of the linear regression of the mouth against the template. We consider the bearded template (Figure 2.2) and the mouth shown in the left part of Figure 2.8, which is the mouth of dr. Böhringer from Figure 1.1. Both the template and the mouth are matrices of size 26×56 pixels. We transform both to vectors and fit the linear regression of the mouth against the template. The vector of residuals transformed back to a matrix is shown in the right part of Figure 2.8. There the mouth can be easily recognized. Grey areas correspond to residuals close to zero. Black or white pixels represent areas with small negative or large positive residuals respectively, in other words the departures from the linear model. These can be found in the lips, moustache and edges in the bottom part, just like in the mouth itself.

In general, the residuals of least squares regression are uncorrelated with the independent variable. In our example the residuals are uncorrelated with the template. They are still strongly correlated with the mouth. The correlation between the mouth and the residuals namely equals 0.73 and between the template and the mouth 0.68. The mouth, the template and also the

residuals have very clear lips and look visually similar to each other.

When we extend the values of both the regressor and the response to the whole interval $[0, 1]$, the residuals attain values between -0.62 and 0.41. The sum of the residuals is clearly equal to zero and we can say that the residuals have a large variability and the appearance of a mouth. Residuals computed in a linear regression of a mouth against a nonbearded template is even more visually similar to a real mouth, namely the lips contain the largest as well smallest residuals.

There seems to be no remarkable heteroscedasticity in the residuals. We have examined plots of residuals against the template, or against fitted values of the response, or against the distance of each pixel from the center of the rectangular area.

The residuals in the linear regression of a mouth against the mouth template are however strongly autocorrelated. Large value of the two-dimensional autocorrelation coefficient computed from the residuals give evidence about it. One such possible coefficient is known as Moran's I and was proposed by Moran (1950). Before we define it, let us consider a matrix $\mathbf{A}$ with m rows and n columns with elements $(a_{ij})_{ij}$. Let us denote the mean of the elements of $\mathbf{A}$ by $\bar{a}$. Moran's I computed from the matrix $\mathbf{A}$ is a general proposal depending on the selection of weights, which describe the way how each element of the matrix $\mathbf{A}$ is compared with its neighbours. This can be done in more ways. Here we compare each pixel with its four neighbours. For such choice the Moran's I can be expressed by the formula

$$I(\mathbf{A}) = \frac{\sum_{i=1}^{m}\sum_{j=1}^{n-1}(a_{ij}-\bar{a})(a_{i,j+1}-\bar{a}) + \sum_{i=1}^{m-1}\sum_{j=1}^{n}(a_{ij}-\bar{a})(a_{i+1,j}-\bar{a})}{\sum_{i=1}^{m}\sum_{j=1}^{n}(a_{ij}-\bar{a})^2}, \qquad (2.6)$$

which does not need the weights any more.

The formula compares each particular element a_{ij} of the matrix $\mathbf{A}$ with elements $a_{i+1,j}$ and $a_{i,j+1}$ and also $a_{i-1,j}$ and $a_{i,j-1}$. Other possible special cases of Moran's I compare each pixel with eight neighbours, namely the same four and also four diagonal neighbours. At any case the definition of Moran's I is a straightforward generalization of the classical autocorrelation coefficient defined for one-dimensional time series. Its interpretation is reasonable only in some contexts, for example in our case when $\mathbf{A}$ is a matrix of grey values corresponding to a two-dimensional image.

The Moran's I of the residuals in Figure 2.8 (right) equals 0.85. The one-dimensional autocorrelation coefficient measured in line segments of the matrix of residuals is also large, for example in a horizontal segment through the lips equals about 0.90. For such data the regression suffers from the so-called R^2-syndrome. This is explained by Greene (1993) in an example of a one-dimensional time series. There the disturbances are autocorrelated, the value of R^2 is therefore overestimated and R^2 does not measure the real relationship between variables.

We believe that there are two reasons, why some areas of the image are correlated with the template without being visually similar to it: the orthogonality-orientation of the correlation (uncorrelated images can appear visually similar) and the large autocorrelation of residuals. While these arguments dishonour the correlation coefficient, these properties are inherited by other correlation measures, such as the weighted correlation or robust analogies of the sample correlation.

Using Residuals as Templates

The residuals can be also used as templates to help locating the mouth. We have considered all mouths from the database of images and fit the linear regression of each one against the bearded

mouth template. Then the mean of such residuals has a similar appearance to the right image in Figure 2.8, which shows residuals obtained from only one given mouth.

In the previous text more templates have been applied at the same time. Each of them was placed on every position of the image and such area was considered suspicious, which has a large correlation with any of the template. Now we want to use the bearded template together with the matrix of residuals as two templates at the same time, but the situation is different.

When a particular area of the image is considered, we fit a linear regression of this area jointly against two regressors, namely the template and the residuals. Then the coefficient of determination is used as a similarity measure between the area and the two templates jointly. This is the case in such of the rows of Table 2.2 in which the residuals are used. The coefficient of determination R^2 increases and the linear fit is improved when the mean residuals are used together with the original mouth template.

We recall that the residuals of the weighted regression are orthogonal to the independent variable in the inner product defined by the weight matrix as proven in Chapter 1.2. Therefore in the weighted regression context we are using the mean of the residuals of the weighted regression. These are however visually very similar to residuals of ordinary least squares.

In Table 2.2 there are results of applying particular templates together with the corresponding mean residuals computed over 124 images. This improves the performance in locating the mouth. After fitting a real mouth against the template and the mean residuals, the new residuals do not contain a lot of structure of a mouth any more. Applying the bearded template from Figure 2.2 together with its residuals brings however improvements only for bearded images, which is not necessary. That is because bearded templates by themselves perform well for all bearded pictures.

We also apply one nonbearded template, the mean of the corresponding residuals and one bearded template. In this case one regression fit is computed for each rectangular area of the image against the nonbearded template and the residuals jointly, and another fit against the nonbearded template itself. We classify such area of the image to be the mouth which has the largest value of R^2 (or R^2_W in the weighted case) over the whole image and over both models.

When the linear regression of a mouth is fitted against the bearded template (Figure 2.2) and corresponding mean residuals together, the residuals are again strongly autocorrelated. For different mouths we have observed the Moran's I defined by (2.6) between 0.80 and 0.85. One-dimensional autocorrelation in the line segment through the lips reaches again 0.90.

The rank correlation r_S of Spearman has a poor performance and a high computational complexity. It is not connected to any regression model and an analogous study of residuals is not possible.

To summarize the paragraph, Table 2.2 shows that none of the methods using the residuals is able to locate the mouth correctly in every image.

Eigenmouths

Principal component analysis is a popular method in the image analysis context, as stated in Chapter 1.1. Computing the principal components for the whole faces has however immense computational requirements. The eigenfaces are eigenvectors of the sample covariance matrix of the data. The number of rows and columns of this square matrix equals to the number of variables, which is in the case of our images $192 \times 256 \approx 50\ 000$ pixels.

We have performed the principal component analysis on mouths from different pictures to discover effects explaining the largest part of the variability of the mouths. We take each mouth

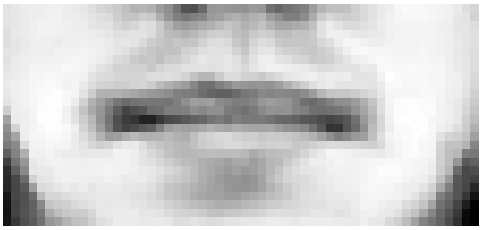

Figure 2.9: The first eigenmouth.

as a matrix with size 26×56 pixels and transform it to a vector of length $26 \times 56 = 1456$ elements. From the 124 mouths from the database we compute the sample variance matrix and its eigenvectors and eigenvalues. The eigenvectors are orthogonal and after being transformed back to matrices of size 26×56 pixels they can be called eigenmouths. Principal components of a given mouth are then defined as linear combinations of the mouth, where the coefficients are given by the eigenmouths.

Eigenmouths corresponding to the largest eigenvalues explain the largest portion of the variability of the mouths. In our case the first eigenmouth explaining 59 % of the variability is shown in Figure 2.9. Other eigenmouths explain only 6 %, 4 %, 3 %, 3 % and none of the remaining ones more than 2 % of the variability.

In the picture of the first eigenmouth we can recognize the lips, the bottom of the nostrils and the bottom corners from the rest. Already the second eigenmouth does not resemble a real mouth very much. The top lip and the whole chin can be visually recognized in it. Other eigenmouths contain lips of various shapes, contrast certain parts of the lips against parts of their neighbourhood or possibly contain a moustache.

We have tried to use the first eigenmouths as templates to locate the mouths in images of the whole faces. Table 2.2 shows that the first eigenmouth performs quite well in this task. With radial weights the mouth is found in 99 % of our images. The second one has almost no ability to locate the mouth. The results remain very similar for eigenmouths computed from all the mouths except for the bearded ones.

Another idea is to use several eigenmouths as templates at the same time. The first two eigenmouths together locate the mouth well in 40 % of images; we do not perform this for the weighted correlation, because the eigenmouths are not orthogonal in the corresponding inner product. A linear combination, which would assign a larger weight to the correlation with the first eigenmouth and a lower weight to the correlation with the second one, does not lead to improvement. Combining the first eigenmouth with some of the previous templates does not bring an improvement.

There is actually no reason for the eigenmouths to perform well as templates, because their purpose is different. The eigenmouths explain the typical differences of particular mouths from the mean mouth. Then each mouth is exactly a linear combination of all eigenmouths and can be reasonably well approximated by a linear combination of several most important eigenmouths. The eigenmouths themselves represent an information complementary to the mean mouth.

Different Size or Rotation

To examine the robustness of the methods to the *size* of the faces we reduce the images from 192×256 to 173×230 pixels, in other words both the height and width of the images is reduced

Table 2.3: Locating the mouth in a smaller or rotated face. The weighted correlation uses radial weights.

Templates (Notation from Table 2.2)	Original		Smaller face		Rotated face	
	r	r_W	r	r_W	r	r_W
1 + 2 + 3 + 4 + 5 + 6 + 7	1.00	1.00	1.00	1.00	0.83	0.92
6	0.91	1.00	0.93	0.98	0.70	0.97
1 + 6	0.94	1.00	0.82	0.91	0.72	0.83
1 + residuals + 6	0.98	1.00	0.91	0.97	0.60	0.91

by 10 %. Also for the Institute of Human Genetics the robustness to the size is important and it is actually more interesting to examine the methods for smaller faces rather than larger faces.

The bearded template (Figure 2.2 left) from the sixth row of Table 2.2 is used, because it is the only template locating the mouth correctly in every image of the standard size. As an example of a nonbearded template we use that from the first row of Table 2.2. Results over the database with 124 pictures are summarized in Table 2.3, together with results obtained with all the seven templates and with three templates, namely the nonbearded template together with its mean residuals and the bearded template.

Table 2.3 shows that the weighted correlation with radial weights performs again better than the sample correlation. A lower number of templates turns out to be more sensitive and the best results are obtained with seven mouth templates.

Next we rotate the pictures by ± 10 degrees and use the same methods without rotating the templates. The percentages of correctly located mouths given in Table 2.3 have been computed for the 124 images rotated by 10 degrees in either direction, in other words for 248 images. The weighted correlation again outperforms the classical one. While the rotation by this angle is rather mild as shown in Figure 4.2, some of the results are considerably worse than those in standard pictures.

In the case of rotated pictures it is not true that more templates ensure a better result. Some templates are more sensitive to the size and rotation than others and other areas than the mouth or eyes can become strongly correlated with one of the templates. One sensitive template can then ruin the results of the whole method and a poor result can be obtained if all available templates are used together. Based on the results we cannot give an unambiguous recommendation which template should be used. A reliable method for locating landmarks in images with a different size and rotation will be described in Chapter 3 using the bearded template and optimal weights.

Chapter 3

Optimization of Templates

Chapter 3 is devoted to optimizing the discrimination between mouths and nonmouths. The weighted correlation coefficient is used as the similarity measure between the template and the image. Firstly we formulate the problem of optimizing the weights while retaining the template. To obtain an approximation to the solution of the complicated nonlinear optimization problem we introduce two algorithms and call them analytical and approximative. We examine the effect and importance of constraints in the form of upper bounds on the weights. While Chapters 3.2 to 3.4 explain the algorithms only on the example of radial initial weights, the results with different initial weights are systematically presented in Chapter 3.5. The aim of the subsequent chapters is to validate the performance of the results in another database of images (Chapter 3.6), to examine a possible preliminary transformation of the data (Chapter 3.7) and to examine the robustness of the methods to different situations such as rotation of the face or nonsymmetry of the mouth (Chapter 3.8). The optimal weights for eye templates are searched for in Chapter 3.9. Finally the optimal template itself is optimized and the optimal weights are retained. This is shown in Chapter 3.10 again on locating the mouth and optimizing of both the template and weights is combined.

3.1 Formulation of the Problem

The optimization of weights for the weighted correlation is a general method not using any special properties of the mouth. Its use is limited neither to mouths nor the context of faces. In the applications below we use the bearded template from Figure 2.2 (left). This was introduced in Chapter 2.3 and locates the mouth in every picture in the available database of 124 pictures when the weighted correlation coefficient is used with radial weights.

We define the Fisher's transform which we apply on the weighted correlations. It is also called z-transformation or inverse *tanh* transformation. Let us denote the weighted correlation between two data vectors $\boldsymbol{x},\ \boldsymbol{y}$ with weights $\boldsymbol{w}$ by $r_W(\boldsymbol{x},\, \boldsymbol{y};\, \boldsymbol{w})$. While its values lie in the interval $[-1, 1]$, the Fisher's transform defined by the formula

$$r_W^F(\boldsymbol{x},\, \boldsymbol{y};\ \boldsymbol{w}) = \frac{1}{2} \log \frac{1 + r_W(\boldsymbol{x},\, \boldsymbol{y};\, \boldsymbol{w})}{1 - r_W(\boldsymbol{x},\, \boldsymbol{y};\, \boldsymbol{w})}$$

extends the values to the whole real line $(-\infty, \infty)$. This monotone transformation is often used for the sample correlation coefficient, because the distribution of the transformed coefficient is

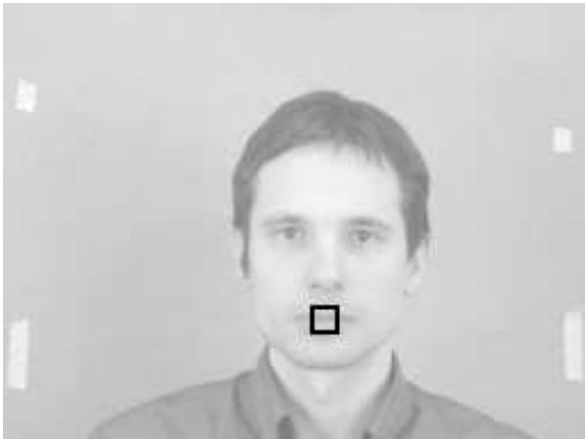

Figure 3.1: The mouth area.

approximately normal with a variance not depending on the correlation itself. The distribution is close to normal already for moderate sample sizes (Hays 1973). Although we have not found references on similar experience with the weighted correlation, we apply the Fisher's transform on it as well because it improves the separation between mouths and nonmouths.

Moreover, Hays (1973) states that the sample correlation coefficient $r(\boldsymbol{x}, \boldsymbol{y})$ is not an interval measure. The distance between sample correlation 0.1 and 0.2 is not the same as the distance between sample correlation 0.8 and 0.9. The Fisher's transform better reflects the position of $r(\boldsymbol{x}, \boldsymbol{y})$ in the collection of all coefficients which then makes different sample correlation coefficients better comparable. This is true also for the weighted correlation coefficient.

Now we come to defining the separation between a particular mouth and nonmouth and then the minimax optimization criterion for the optimal weights. We need to define the concept of mouth and nonmouth precisely. Let us look at Figure 3.1, which shows the boundary of the *mouth area*. This is constructed for every picture as the rectangle of the size 11×11 pixels centered in the midpoint of the mouth.

When we place a certain mouth template on every possible position in Figure 1.1, the template is each time compared with a rectangular area, which has the same size as the template. The rectangular area is considered to belong to the *mouth*, if its midpoint belongs to the mouth area defined above and shown in Figure 3.1. All other areas with the midpoint outside of the mouth area will be called *nonmouths*. As Figure 3.1 shows the mouth area contains the middle parts of the lips, but reaches neither the nostrils nor other landmarks.

Let us now consider a particular picture and a particular mouth and nonmouth in it. The template, initial weights, mouth and nonmouth must have the same size. In our application with the bearded template (Figure 2.2 left) these matrices have the size 26×56 pixels. Let the matrices be transformed to vectors of length $n = 26 \times 56 = 1456$ pixels.

We use the notation $\boldsymbol{x} = (x_1, \ldots, x_n)^T$ for the vector of grey values of the mouth, $\boldsymbol{z} = (z_1, \ldots, z_n)^T$ for the nonmouth, $\boldsymbol{t} = (t_1, \ldots, t_n)^T$ for the template and $\boldsymbol{w} = (w_1, \ldots, w_n)^T$ for the weights. Then we consider the function $f(\boldsymbol{x}, \boldsymbol{z}, \boldsymbol{t}, \boldsymbol{w})$ comparing the weighted correlations

after the Fisher's transformation in the form

$$f(\boldsymbol{x}, \boldsymbol{z}, \boldsymbol{t}, \boldsymbol{w}) = \frac{r_W^F(\boldsymbol{x},\, \boldsymbol{t};\, \boldsymbol{w})}{r_W^F(\boldsymbol{z},\, \boldsymbol{t};\, \boldsymbol{w})}. \tag{3.1}$$

This is the separation between the mouth $\boldsymbol{x}$ and nonmouth $\boldsymbol{z}$ in a particular picture using the template $\boldsymbol{t}$. The value of (3.1) should be large in order for the mouth to be located reliably. The value larger than 1 means that the mouth classified correctly and thus discriminated from the nonmouth.

Now we come to the overall optimization criterion. The idea is to find such weights which separate well the mouth from every nonmouth in a particular image, even from the nonmouth with the largest similarity to the template. Therefore in a particular picture we consider the maximum of the numerator of (3.1) over all positions of the mouth and the minimum of the denominator over all nonmouths in the picture. This will be now explained more precisely.

Keeping in mind that every area with the appropriate size (here 26×56 pixels) with its midpoint in the mouth area is considered to be the mouth, we find the mouth with the largest weighted correlation with the mouth template in every image. Such mouth will be simply called the best mouth. During the computation the coordinates of the midpoint of the mouth are used which have been located automatically by the methods of Chapter 3.

The position of the mouth in every image must be known and it is important that it is known precisely. In general these can be either located manually or by software.

We also find the nonmouth which has the weighted correlation with the mouth template larger than any other nonmouth. We call it the worst nonmouth, because it is the worst one with respect to the discrimination from the mouth. Using the notation of above, we compute in every image

$$\min_{\text{nonmouths}} \max_{\text{mouths}} f(\boldsymbol{x}, \boldsymbol{z}, \boldsymbol{t}, \boldsymbol{w}) \tag{3.2}$$

using the given weights.

To control the performance of the best nonmouth over the whole database of images, we can find the worst case. This is defined over the whole database $\mathcal{I}$ of images and over all nonmouths $\boldsymbol{z}$ and all positions of the mouth $\boldsymbol{x}$ in every particular image $i \in \mathcal{I}$ as

$$\min_{i \in \mathcal{I}} \min_{\boldsymbol{z} \in \mathcal{I}} \max_{\boldsymbol{x} \in \mathcal{I}} f(\boldsymbol{x}, \boldsymbol{z}, \boldsymbol{t}, \boldsymbol{w}) = \min_{\text{images}} \min_{\text{nonmouths}} \max_{\text{mouths}} f(\boldsymbol{x}, \boldsymbol{z}, \boldsymbol{t}, \boldsymbol{w}). \tag{3.3}$$

The overall optimization criterion for the weights consists in maximizing (3.3) over the weights.

The bearded template with the radial weights (Figure 2.4) locate the mouth in all 124 images. The value of (3.3) equals 1.11. The radial weights will be used as initial in the optimization. Equal weights namely locate the mouth only in 94 % of images and the worst separation (3.3) attains only 0.78.

Here we stress that a ratio of two weighted correlations (after the Fisher's transform) is optimized rather than just one weighted correlation itself. A large value of the weighted correlation between the template and the mouth does not imply the nonmouths to have a lower correlation with the template.

The weight in each pixel is an unknown parameter. This is a nonparametric approach and the optimization is carried out in a space of a high dimension. The overall optimization problem is highly nonlinear and there is no guarantee that the global maximum will be found. In the following chapters we describe two approaches for the approximation to the optimal weights.

The first one is based on the first order approximation to the separation (3.1). The second one is simpler, modifying weights in pixels which are selected at random. Although an analytical approach should be in general more precise, we will later see that both methods approach the problem from a different perspective and finally we obtain the best results by combining both methods together. Standard approaches to optimization are discussed at the end of Chapter 3.5.

Both the analytical and approximative methods of Chapters 3.2–3.6 are general, not designed only for mouths. We require the solution to be symmetric. This may be an advantage for the mouths, however may not be suitable for other landmarks. In the literature we have not found a similar optimization of the discrimination between images.

3.2 Analytical Search

The linear approximation is one possibility for maximizing (3.3) over the weights. Let us consider the worst case over the whole training set of images, namely the mouth and worst nonmouth from the same image. We introduce the notation $\boldsymbol{x}$ for the mouth and $\boldsymbol{z}$ for the nonmouth. Further we consider a given template $\boldsymbol{t}$ and initial weights $\boldsymbol{w}$. We assume that the matrices $\boldsymbol{x}, \boldsymbol{z}, \boldsymbol{t}$ and $\boldsymbol{w}$ have the same size and let them be transformed to vectors.

The function $f(w_1, \ldots, w_n;\ \boldsymbol{x}, \boldsymbol{z}, \boldsymbol{t})$ now denotes the separation (3.1) for the worst case over all images. The first order approximation or Taylor's expansion for the separation

$$f(w_1 + \delta_1, \ldots, w_n + \delta_n;\ \boldsymbol{x}, \boldsymbol{z}, \boldsymbol{t}) \approx f(w_1, \ldots, w_n;\ \boldsymbol{x}, \boldsymbol{z}, \boldsymbol{t}) + \sum_{i=1}^{n} \delta_i \frac{\partial f(w_1, \ldots, w_n;\ \boldsymbol{x}, \boldsymbol{z}, \boldsymbol{t})}{\partial w_i} \quad (3.4)$$

can be considered, because f is a differentiable function of the weights. It is quite straightforward to compute the partial derivatives and we do not give the lengthy formula here. The Taylor's theorem informs about the magnitude of the remainder term of the linear approximation. The second derivatives in this case exist and therefore the remainder term is asymptotically negligible if the size of the constants $\delta_1, \ldots, \delta_n$ tends to zero. The second derivatives are already too complicated formulas and we use only the approximation of the first order in the Taylor's expansion.

The aim is to modify the weights to improve the separation. We add small constants $\delta_1, \ldots, \delta_n$ to the initial weights. To improve the separation between the mouth and nonmouth, the left side of (3.4) should be increased. Therefore the sum on the right side of (3.4) is desirable to be large.

We solve the linear problem

$$\max_{\delta_1, \ldots, \delta_n} \sum_{i=1}^{n} \delta_i \frac{\partial f(w_1, \ldots, w_n;\ \boldsymbol{x}, \boldsymbol{z}, \boldsymbol{t})}{\partial w_i} \quad (3.5)$$

with variables $\delta_1, \ldots, \delta_n \in \mathbb{R}$ under the constraints

- $\sum_{i=1}^{n} \delta_i = 0$
- $0 \leq w_i + \delta_i \leq c$ with a certain c, $\quad i = 1, \ldots, n$
- $-\varepsilon \leq \delta_i \leq \varepsilon$ with a certain ε, $\quad i = 1, \ldots, n$
- the weights must be symmetric (along the vertical axis in the middle).

The first constraint states that the sum of the new weights will be the same as the sum of the original weights, namely 1. The next set of constraints requires the new weights to be nonnegative and bounded by a given upper bound c. This a prevention against highly influential weights. The third set of constraints allows the weights to be modified only slightly compared to the previous weights.

The second and third sets of constraints can be formulated as

$$\delta_i \leq \min\{\varepsilon,\ c - w_i\}, \quad i = 1, \ldots, n$$

and

$$\delta_i \geq \max\{-\varepsilon,\ -w_i\}, \quad i = 1, \ldots, n.$$

The latter one is equivalent to

$$-\delta_i \leq \min\{\varepsilon,\ w_i\}, \quad i = 1, \ldots, n.$$

Bounding the weights is a prevention of degenerating. For a particular mouth and nonmouth it is typically possible to find degenerated weights which improve the separation drastically.

In the search for the mouth, we require the weights to be symmetric along the vertical axis through the middle. It is possible to include the symmetry conditions in the linear program. Such computation is then unnecessarily slow, because it requires to have as many of the variables $\delta_1, \ldots, \delta_n$ as there are pixels in the mouth template.

The following approach considers only a half of the variables for the linear problem and thus reduces the computational complexity. Let us explain it for a template with an even number of columns. For the symmetric case one half of the number of the weights represents unknown parameters and the remaining weights are equal to their symmetry counterparts. The derivatives are not symmetric, because the mouth and nonmouth are not symmetric. In the objective function (3.5) we multiply each variable by the sum of the partial derivatives in the corresponding pixel with the partial derivative in the symmetry counterpart of the particular pixel. Then the number n in the linear problem (3.5) and in the later text is only a half of the number of the pixels in the template.

The requirement of symmetry is in general not needed. It is reasonable in the search for the mouth, because the mouths in the images are also approximately symmetric. A nonsymmetric optimal solution would depend too much on the nonsymmetry of the several worst cases (mouths and nonmouths) which is not desirable. The requirement of symmetry would not be suitable for locating nonsymmetric landmarks.

The linear problem for the separation in the worst case is solved many times during the iterative algorithm. As the worst separation increases, sooner or later it reaches the level of the second worst case. Then we need to propose such new weights which improve the separation for both cases simultaneously. Otherwise a solution for one case would not necessarily improve the other case.

We will introduce another constraint or rather a set of constraints for the situation with several worst cases at the same time. These will have the separation slightly better than the very worst case rather than the very same separation. Let us therefore consider all nonmouths together with the mouths from the same images, which have the separation larger than the very worst case but not more than by 0.01. All these will be included in the optimization. Omitting one of them could propose a modification of the weights, which would decrease the separation for this omitted case and the algorithm could not iterate further.

For each one of the worst cases described in the last paragraph let us denote the mouth by $\boldsymbol{x}^*$ and the nonmouth by $\boldsymbol{z}^*$. Then

$$\sum_{i=1}^{n} \delta_i \frac{\partial f(w_1, \ldots, w_n;\ \boldsymbol{x}, \boldsymbol{z}, \boldsymbol{t})}{\partial w_i} = \sum_{i=1}^{n} \delta_i \frac{\partial f(w_1, \ldots, w_n;\ \boldsymbol{x}^*, \boldsymbol{z}^*, \boldsymbol{t}^*)}{\partial w_i} \tag{3.6}$$

is a set of constraints optimizing the very worst case together with such cases which are only slightly better than the worst one.

The values of $\delta_1, \ldots, \delta_n$ can be expected only small and it indeed turns out that they must be bounded by very small values of ε. The reason is the first order approximation which may work only in a small neighbourhood of the initial weights.

We use **linear programming** to solve the linear problem. We have implemented the simplex algorithm in the computer language C and the subroutine is fast and able to handle large data input.

To formulate the linear problem in the canonical form (Sultan 1993) we must introduce nonnegative variables $\delta_1^+, \delta_1^-, \ldots, \delta_n^+, \delta_n^-$ corresponding to the positive and negative parts of the original variables, namely $\delta_i = \delta_i^+ - \delta_i^-$ for each $i = 1, \ldots, n$.

The linear problem

$$\max_{\delta_1^+, \delta_1^-, \ldots, \delta_n^+, \delta_n^-} \sum_{i=1}^{n} (\delta_i^+ - \delta_i^-) \frac{\partial f(w_1, \ldots, w_n;\ \boldsymbol{x}, \boldsymbol{z}, \boldsymbol{t})}{\partial w_i} \tag{3.7}$$

is to be solved under the constraints

1. $\sum_{i=1}^{n} (\delta_i^+ - \delta_i^-) = 0$
2. $\delta_i^+ - \delta_i^- \leq \min\{\varepsilon,\ c - w_i\}, \quad i = 1, \ldots, n$
3. $-\delta_i^+ + \delta_i^- \leq \min\{\varepsilon,\ w_i\}, \quad i = 1, \ldots, n$
4. one or more conditions corresponding to (3.6) in the form

$$\sum_{i=1}^{n} (\delta_i^+ - \delta_i^-) \left(\frac{\partial f(w_1, \ldots, w_n;\ \boldsymbol{x}, \boldsymbol{z}, \boldsymbol{t})}{\partial w_i} - \frac{\partial f(w_1, \ldots, w_n;\ \boldsymbol{x}^*, \boldsymbol{z}^*, \boldsymbol{t}^*)}{\partial w_i} \right) = 0$$

5. $\delta_i^+ \geq 0,\ \delta_i^- \geq 0, \quad i = 1, \ldots, n.$

The perfect canonical form requires nonnegative variables δ_i^+ and δ_i^- for each $i \in 1, \ldots, n$. All other constraints must be in the form of equalities. Therefore there must be dummy variables introduced for the second and third constraints. The right hand sides of the equalities are nonnegative which is also a requirement for the perfect canonical form.

To find the initial basis for the simplex algorithm we introduce an auxiliary basic variable for the first constraint of above. A usual approach is to minimize the value of this variable (but the objective function must formulated only in terms of nonbasic variables) by the simplex algorithm; as this iterates, the auxiliary variable becomes nonbasic. However in general it still can happen that the auxiliary objective function is zero, the algorithm cannot run further and the variable remains basic. This happens exactly in this linear problem. While the standard method for selecting the pivot does not help, Sultan (1993) states that it is possible to pivot

on any nonzero coefficient in the row of the auxiliary variable and this finds the initial basis for the original problem. The auxiliary variable can be omitted and the linear problem is solved by repeating the steps of the Gaussian elimination.

Additional constraints corresponding to (3.6) are also in the form of equalities. If these are present, they represent additional rows in the matrix of the linear problem and there must be auxiliary variables corresponding to them introduced as well. In the same way the simplex algorithm is used to find a basis not including these variables, they turn out to equal zero and can be deleted. Then the simplex algorithm is used to solve the original problem and the only difference consists in additional rows in the matrix.

So far we have described one step of the algorithm, which will be repeated iteratively. Let us now describe the algorithm as a whole.

- Take a certain template.
- Begin with symmetric initial weights.
- Find the worst case over all images. That is the mouth and nonmouth from one particular image with the smallest separation. Find such mouths and nonmouths, which have the separation larger than the worst case only by 0.01 or less.
- Compute the partial derivatives, formulate the linear problem and use linear programming to solve it.
- Add the solution $\delta_1, \ldots, \delta_n$ to the weights $w_1, \ldots, w_n$ to obtain a proposal for new weights $w_1 + \delta_1, \ldots, w_n + \delta_n$.
- If there is an improvement in the criterion (3.3) over all images and over all nonmouths, formulate again the linear program for the worst case etc.

Now we give only a few technical details about the algorithm and its implementation which are worth to be mentioned. It turns out that the upper bound ε must be very small in order for the algorithm to improve the separation. However the smaller value of ε, the slower the computation is. Therefore our program chooses a larger ε at first and if this does not give an improvement, the procedure with a smaller choice of ε is carried out.

The new weights are used for two different things. The first one is to check the improvement in the overall separation (3.3). If there is an improvement, the second application is to find the worst case or worst cases over the whole database. The speed of the algorithm is improved if both steps are computed at the same time. For the linear program we need also worst cases slightly better than the very worst one; however the worst separation is not known yet as the worst cases are searched for. Therefore we find such nonmouths which have separation from the mouth less than by 0.01 better than the very worst separation obtained with the previous version of the weights.

To improve the speed we work only with 10 images in which the value (3.2) is the smallest. We expect that the worst nonmouths with the new weights will appear also in these images. Only when the value of the worst separation over the whole training set is larger than the initial one by 0.1, we detect again the 10 images with the smallest value (3.2). At the beginning of the computation it is enough to work with 10 worst images, later in the course of the algorithm we consider as many as 20 worst images.

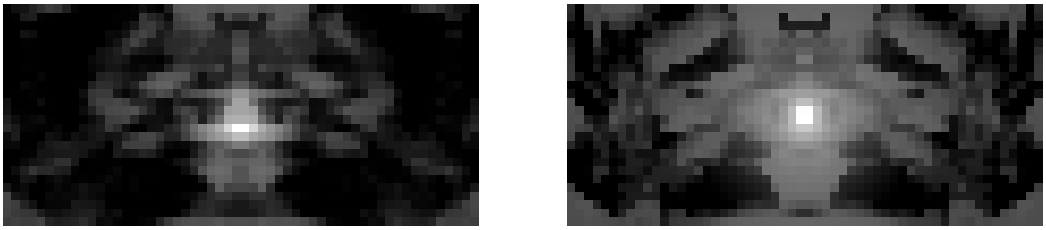

Figure 3.2: Weights for the bearded mouth template. Solution of the analytical search with different values of the upper bound c. Left: $c = 0.005$. Right: $c = 0.02$.

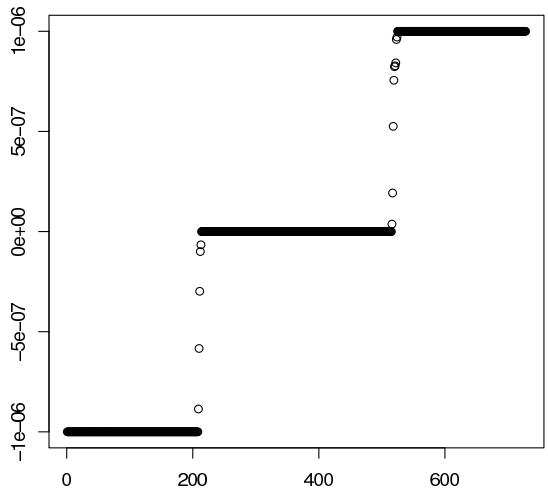

Figure 3.3: Sorted values of the solution of the linear problem.

It would be also possible to store only the mouth and nonmouth in each of the 10 worst images. However after modifying the weights the worst nonmouth can be shifted from the previous position by one pixel and therefore it is advisable to search for the worst cases each time.

Results of the analytical search

We use the bearded template and the radial weights and apply the analytical search to improve the discrimination between mouths and nonmouths in the database with 124 images. The template is a matrix with 26 rows and 56 columns, which makes $26 \times 56 = 1456$ pixels altogether.

The worst separation with radial weights equals 1.11. However we require the resulting weights bounded from above by $c = 0.005$. Because the radial weights exceed this bound in 4 pixels in the midpoint, we have set the weights in these pixels to 0.005 and multiplied the remaining by a suitable constant to obtain $\sum_{i=1}^{n} w_i = 1$. These transformed weights with the bearded template again find the mouth correctly in each of the 124 images of the database and the worst separation equals 1.13.

First of all let us describe the derivatives. They are very diverse for different cases, for different mouths and nonmouths. They produce antagonistic effects on some of the weights: some cases require the weights in particular pixels to decrease in order to improve the separation, others require to decrease them. In general the optimal weights represent a complicated compromise over the worst mouths and nonmouths. If we visualize them as a matrix of the

size 26×56 pixels, there can be parts of the lips from the template recognized and remarkable parts of the nonmouth as well. If for example the nonmouth contains dark hair and lighter neighbourhood, it is possible to recognize these two parts and a clear boundary between them in the derivatives too. In general the derivatives reflect specific properties of the mouth and mainly nonmouth, for which they are computed.

There are 1456 weights for the weighted correlation. Using the symmetry one half of them are unknown parameters. Then the value of n in the linear problem (3.7) and in the following constraints equals $n = 1456/2 = 728$. The matrix in the linear problem contains $2n + 1 = 1457$ rows and possibly several other rows corresponding to other nonmouths, which have the separation only slightly better than the worst one. There happen to be as many as 10 worst nonmouths from different images during the computation. Some of them appear in the linear program several times, because the nonmouth shifted for example by one pixel aside is already a different case and can have a low discrimination from the mouth as well. The number of columns of the matrix is $4n + 1 = 2913$. In the search for the initial basis there are also several additional columns for auxiliary variables for possible other worst nonmouths, which have the separation only slightly better than the worst one. The matrix of the linear program thus contains more than 4 millions elements and it takes about 4 minutes to solve it with our program.

The matrix of the linear program contains many zero elements. It consists not only of diagonal blocks; it further contains one row corresponding to the constraint that the sum of the variables equals 0. And possibly several other rows corresponding to cases, which have the separation only slightly better than the very worst case. In these rows there can be any real values, which depend on the partial derivatives of the separation (3.1). In the course of the simplex algorithm the structure of the diagonal blocks dissolves and we have found neither any special structure (sparseness) in the matrix nor a tailor-made approach for solving the problem more easily and quickly.

The output of the linear program as a vector is denoted by $\delta_1, \ldots, \delta_n$. Its example is shown in Figure 3.3. In this case $c = 0.005$ and $\varepsilon = 0.000\ 001$ and the optimization took place over four different worst cases from different images. About 30 % of the resulting values attain ε and a similar number of pixels $-\varepsilon$. Most of the remaining weights are zero, mainly because of the constraints (3.6). Some other weights correspond to w_i or $0.005 - w_i$, which are the bounds from constraints of (3.7) for certain index i.

In the example of Figure 3.3 the corresponding derivates transformed to a matrix suggest to increase 30 % of the weights and decrease a similar number of them. Very roughly speaking it is suggested to increase the weights in the lighter pixels of the left image in Figure 3.2 and to decrease the weights in the black areas. This tendency more or less retains as the algorithm iterates to the solution.

The resulting weights with the upper bound $c = 0.005$ improve the separation from 1.13 to 1.68. These are shown in the left part of Figure 3.2. The largest weights are shown as white and these are situated in the middle area of the rectangle. The upper bound is attained only in two pixels in the midpoint of the rectangle, which is inherited from the initial weights. Black pixels then correspond to pixels with zero or very low weights. Almost 20 % of the weights are equal to zero. We note that optimal weights should not resemble the template. Namely we expect larger weights in the lips but smaller weights for example in the moustache.

The computation of this result takes about 24 hours. It is necessary to proceed in very small steps, with small values of ε. At the beginning $\varepsilon = 0.000\ 020$ can be used but later we can proceed only with smaller values. The computation could run further with extremely small

values of ε which would then improve the result in the worst case only slightly. We stop the computation in the moment when further improvements are possible only with ε smaller than 0.000 001.

Now we present results of the analytical search with a different value of the upper bound c on the weights. The result obtained with $c = 0.02$ again starting with radial initial weights is shown in the right image of Figure 3.2. The worst separation over the whole training set of 124 images then equals 1.65. About 41 % of the weights equal exactly zero. The largest weights in the midpoint have the value 0.0096.

Another possibility is to choose $c = 0.01$. The solution is identical with that obtained with $c = 0.02$, because this upper bound is not attained in the resulting weights. Also the analytical search without any upper bound c would yield the same result. The weights of the solution of the analytical search cannot become very large because the weights are in each step modified by very small constants $\delta_1, \ldots, \delta_n$. When these are not very small, it turns out that the proposed modification of the weights does not improve the overall worst separation.

The resulting weights obtained with $c = 0.005$ and $c = 0.02$ with radial initial weights have been described and will be compared now. We recall that both are shown in Figure 3.2 and the worst separation obtained with them is 1.68 and 1.65, respectively. Using the larger upper bound $c = 0.02$ the optimization takes place in a larger space of possible solutions and the real optimum must give a larger worst separation. However we use only an approximative algorithm which seemingly does not come to the global extreme. The approach with $c = 0.02$ proceeds quickly to a local extreme, where the solution also remains and the solution with the stricter upper bound $c = 0.005$ overcomes its results. Also the algorithm does not allow the large weights in the midpoint to be reduced substantially. This reveals the difficulty of this nonlinear optimization problem in a high dimension.

The largest weights on the right of Figure 3.2 with $c = 0.02$ are larger than the largest weights on the left, but the value of the upper bound c influences mainly the midpoint, otherwise the solutions are not so different. It plays a role that in the search with $c = 0.005$ we have made the initial weights smaller in the midpoint before starting the algorithm so that the initial weights do not exceed 0.005. The remaining weights were than enlarged to make the sum of the weights equal to 1. This transformation itself helped and prevented the solution from coming to the local extreme which was obtained without this initial transformation.

When the solution of the analytical search is used as weights for locating the mouth with the bearded template, the worst nonmouths over the whole database are symmetric, for example one of the eyes or more typically the nose; the midpoint of the most suspicious area is in this case in the middle between the nostrils. Nonsymmetric nonmouths report a smaller weighted correlation with the template. This is a consequence of limiting the resulting weights to be symmetric.

Now we come back to the example of Chapter 3.2. We recall that Figure 2.5 shows a nonmouth which has a larger sample correlation with the bearded template (Figure 2.2) than the mouth of the same image. That pair is one of the worst cases with equal weights and we have observed in Chapter 3.2 that radial weights discriminate the mouth from the nonmouth very well.

Table 3.1 presents the weighted correlation between the mouth and the template and between the nonmouth and the template from this example with different choices of weights. Using the solution of the analytical search with $c = 0.005$ and starting with radial weights, the weighted correlation between the mouth and the template now attains 0.67 and between the nonmouth

Table 3.1: Results of the example with the mouth and nonmouth from Figure 2.5.

	Weights		
	Equal	Radial	Analytical
r_W(mouth, template; $\boldsymbol{w}$)	0.48	0.66	0.67
r_W(nonmouth, template; $\boldsymbol{w}$)	0.52	0.38	0.39
$\frac{r^F_W\text{(mouth, template; } \boldsymbol{w})}{r^F_W\text{(nonmouth, template; } \boldsymbol{w})}$	0.90	1.97	2.00

and the template 0.39. The separation between the mouth and nonmouth is therefore 2.00, which is the ratio of the two values after performing the Fisher's transform on both of them. This mouth and nonmouth do not represent the worst pair obtained with the radial weights. Therefore the results of the analytical search are not so much better than those obtained with the radial weights, which yield the separation between the mouth and nonmouth 1.97.

Using equal weights the nonmouth is more correlated with the template than the mouth and therefore the separation is lower than 1, namely 0.90. The example convicts the sample correlation not to be a suitable similarity measure.

Optimization of weights without the symmetry requirement is much slower because of a twice larger number of variables. We have not computed the solution. In general the worst separation can be expected to be larger. Such approach would however use too specific properties of the training images.

Other Possible Formulations of the Problem

Solving the linear problem is time consuming. Now we discuss if such approach is really necessary or there exists another way of finding the solution.

Let us now try to propose the new weights very simply. Let the constant ε be chosen as above. The proposal of a new weight w_i for a certain index i can add either ε or $-\varepsilon$ to the weight w_i based on comparing the derivative with respect to the corresponding weight

$$\frac{\partial f(\boldsymbol{w};\, \boldsymbol{x}, \boldsymbol{z}, \boldsymbol{t})}{\partial w_i}$$

with the median of all these derivatives over all indexes. Further let us consider another mouth $\boldsymbol{x}^*$ and nonmouth $\boldsymbol{z}^*$ which has the worst separation not better than the very worst case by more than 0.01. This is the same notation as in (3.6). Now there are two sets of derivatives to be taken into account, the first ones are evaluated for the mouth $\boldsymbol{x}$ and nonmouth $\boldsymbol{z}$ of the worst case and the others for $\boldsymbol{x}^*$ and $\boldsymbol{z}^*$. We propose the new weights as the initial ones plus the sum of two indicators, where the first one is the same as above and the other one compares the partial derivative with respect to w_i evaluated in $\boldsymbol{x}^*$ and $\boldsymbol{z}^*$ with the median of these derivatives over all indexes. In an analogous way the weights can be proposed for a larger number of worst cases. Such proposal in general turns out not to improve the worst separation reliably.

Another possible choice is to start with initial weights and add directly the derivatives multiplied by a small constant to them. This allows the weights to be changed more substantially in certain pixels. However there is no clear modification for the situation with several of the worst mouths and nonmouths. The difficulty of the optimization problem is in the additional

worst cases formulated by (3.6) and we can summarize that the only reasonable results have been obtained really by solving the linear problem with the simplex algorithm, which is a slower and more complicated method than the other proposals of this paragraph.

3.3 Approximative Search Without Constraints

While Chapter 3.2 applies the linear approximation to the nonlinear overall optimization problem, this chapter proposes a simpler and rougher approach which is at the same time quite powerful. This approximative approach will be now called without constraints, because we do not require additional constraints in the form of an upper bound on each of the weights. A constrained version will be presented later in Chapter 3.4.

Just like in Chapter 3.2 the method starts with a certain template and initial weights. The weights are improved in iterations over the database of images, which can be understood as a training set. The worst case is the mouth and nonmouth from the same image, which have the worst separation over the whole database (3.3). The value of (3.2) is then computed in every image. We change the weights slightly and check if the value of (3.3) increases. Then the new weights can be accepted and the procedure repeated. When the new weights do not lead to an improvement, the previous weights are retained. Then a new change of the weights in a different way can be proposed, again the condition for improvement is checked and this algorithm keeps iterating.

The main idea is to change the weight in pixels selected at random. The initial radial weights are symmetric along the virtual vertical line in the middle and we require also the resulting weights to be symmetric. One pixel in the left half of the template is selected at random. The weight is modified in this pixel and in the same way in its symmetry counterpart in the right half of the template. There are more possibilites how to change the weights, at any case the sum of the initial weights is exactly one and this property must be retained in the new weights in each iteration as well. Therefore changing the weights in two pixels must be followed by a further change of the remaining pixels to obtain $\sum_{i=1}^{n} w_i = 1$.

We add a constant to the weights in the two pixels, namely 0.001. All of the remaining weights are then multiplied by a constant such that the sum of all weights becomes again equal to 1. The worst separation (3.3) is then computed and compared with the previous value. If it increases, the new weights are kept and we try to modify the weights again in the same two pixels.

If increasing the initial weights in the two pixels does not give an improvement, we can decrease the weights by a constant (again 0.001) or set the weight directly to zero not to obtain a negative weight. Then again the separation (3.3) is computed and the new weights are retained if and only if the value of (3.3) increases compared with the previous value.

A faster and rougher version of the algorithm can double the weights in the two pixels or on the other hand to set them directly to zero. The resulting worst separation must be lower than before, however the difference turns out not to be big.

It is possible to proceed in a systematic way one pixel after another rather than to select pixels at random. We have observed the results to depend heavily on the initial position and direction of the progression. We have considered for example the pixels in rows one after another, starting at the left bottom corner moving towards the symmetry axis and in rows going upwards. The same procedure through the rows starting in the top yields very different results, namely the pixels near the start obtain large weights. Therefore we consider the random choice of pixels

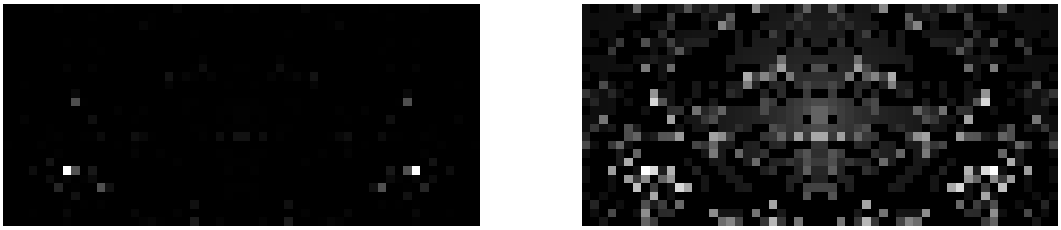

Figure 3.4: Left: solution of the approximative search without constraints. Right: the same image in the log scale.

Figure 3.5: Largest weights from Figure 3.4 and their positions in the template.

to be more reliable.

The pixels can be selected with or without replacement. In general a repeated selection of pixels can yield a different result. As the algorithm iterates, suitable weights in certain pixels are found. Later other pixels are examined and their weights changed. Then returning again to pixels, which have already been considered, can change their weights in a different way than before and further improve the results. However we have observed such improvement in the worst separation to be only marginal and therefore we select each pair of pixels only once.

The method is computationally very intensive. Each time new weights are proposed and directly used to find the best mouth and nonmouth in every image and then the separation in the worst case is compared with the previous result. Locating the mouth with the new version of weights does not need more than a few seconds in one image, but is repeated many times over the whole training set of images. To improve the algorithm we have used relevant suggestions of Chapter 3.2 and moreover the following idea.

Our implementation of the algorithm stores the 10 worst images ordered with respect to the worst separation with the previous version of weights. The worst image is then checked as the first one with the new weights. As soon as the worst separation with new weights in any image is lower than the worst separation with the previous weights over the whole database, locating the mouth with the new weights stops, the previous weights are retained and we proceed to a new proposal to modify the weights. This saves time by not examining all 10 images when the worst ones are examined at first.

Results of the approximative search without constraints

The approximative search was applied to improve the discrimination between mouths and nonmouths in the database with 124 images. We retain the bearded template from Figure 2.2, start with radial weights and optimize the weights for the weighted correlation. We require the solution to be symmetric. The computation requires about 10 hours.

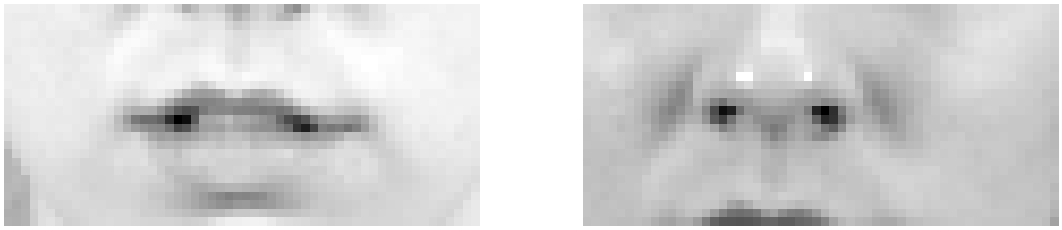

Figure 3.6: The worst case with radial weights over the whole database of 124 images. Mouth (left) and nonmouth (right) from the same image.

The left image of Figure 3.4 shows the solution of the unconstrained approximative search and the right image shows the same weights in the log-scale. While the initial separation in the worst case attains 1.11, it is now improved to 2.10. There are two remarkable white pixels with highly influential weights, which are equal to 0.13. There are three other pairs with weights about 0.04 in each pixel.

The positions of the most influential pixels are shown in the template in Figure 3.5. They can be found in grey areas of the face, downwards from the mouth towards the corners of the rectangle. One of the pairs of pixels with weights about 0.04 is located just next to the most influential points, another such couple not very far from there and another is located near the boundary of the moustache. The sum of the weights in these 8 pixels equals about 0.50, while the sum of all the weights equals 1. About one half of the information of the weights is therefore contained in a very small number of pixels.

While the plot of the weights themselves shows only the influential points, the plot in the log scale (right image of Figure 3.4) reveals better the structure of the smaller weights. In the bottom of the picture there can be larger weights forming a part of a parabola can be recognized. That structure looks like the boundary of the bottom jaw and the chin, but actually corresponds to a boundary between a lighter part of the jaw and chin and a darker area in the corners and the bottom. Such vague boundary be seen also in the template (Figures 2.2 left and also 3.5). Larger weights also correspond to the boundary of the moustache. There is not much weight in the lips and the moustache. The bottom parts of the moustache have zero weights and its top parts small positive weights. Altogether 46 % of the weights equal exactly zero.

The value of the worst separation (3.3) with these weights and the bearded template equals 2.10 in an image of one lady. The mouth is in this case perfectly localized in its midpoint. The worst nonmouth contains eyebrows and reaches the hair and top of the ear. The weighted correlation between the mouth and the template attains 0.60 and between the nonmouth and the template 0.32. The separation between them is the ratio of the two weighted correlations, on which the Fisher's transorm is applied, namely 2.10. There are seven other images with the worst separation between 2.10 and 2.11.

The solution of the approximative unconstrained search improves the discrimination among mouths and nonmouths dramatically not only for the worst case over the whole training set of images. Taking the value largest value of (3.2) in every image and then taking the median over all images, the median of the worst separations with radial weights equals 1.61 and with the weights from Figure 3.4 equals 2.66.

Discussion of the Results

In this section we discuss the results of the unconstrained approximative search and explain the purpose and effect of highly influential weights. Firstly let us describe the worst cases for different choices of weights, namely equal and radial ones and the weights from Figure 3.4 obtained from radial initial weights. Each of the worst cases is a mouth together with the worst nonmouth, which both come from the same image. The mouth is typically well positioned, the midpoint of the lips corresponds to the midpoint of the template. Big differencies among the worst nonmouths are noticeable when different weights are used.

Using equal weights the worst nonmouths are not visually similar with a mouth. One of the worst cases is shown in Figure 2.5, where the nonmouth is a part of the shirt. Other examples include rather homogeneous areas in the hair or shirt. The sample correlation is scale-invariant and also almost homogeneous areas in the background can have a large correlation with the template.

Radial weights make the middle area more important. The worst nonmouths are then areas with a dark middle and many of them are not symmetric. Most often these are areas centered in one of the nostrils. The worst mouth and nonmouth for radial weights are shown in Figure 3.6.

The optimization of weights improves the separation for such cases, which are the worst for the radial weights. In general large weights correspond to the pixels, which contribute the most to the discrimination between mouths and nonmouths. They can disqualify nonsymmetric images (nonmouths) from being highly correlated with the mouth template. The weights obtained with the unconstrained approximative search (Figure 3.4) report different worst cases: some of the worst nonmouths are nonsymmetric such as areas centered in one nostril, others are symmetric such as the eye or the chin.

Now we try to explain the existence and effect of the highly influential weights in the solution of the unconstrained approximative search (Figure 3.4) in one particular example, which considers the mouth and nonmouth from Figure 3.6. The sample correlation between the mouth and the bearded template equals 0.50 and between the nonmouth and the template equals 0.38. The separation between the mouth and nonmouth equals therefore 1.36. Using radial weights the weighted correlation between the mouth and the template equals 0.61 and between the nonmouth and the template 0.56. The separation between the mouth and nonmouth equals 1.11. That is the smallest value of the separation over the whole training set of images.

We have transformed the template, mouth and nonmouth to have the values in the whole interval $[0, 1]$. The following explanation will resemble that of Chapter 3.2. In Figure 3.7 on the left we show the plot of grey values of the mouth against the bearded template and on the right the plot of the nonmouth against the same template. The weighted correlation is based on goodness of the weighted regression fit and compares this with the submodel given by the weighted mean of the response. The red lines correspond to the weighted regression and the blue horizontal lines to the weighted means, each time with radial weights. The lips are the most important part of the template and their low grey values are shown in the left part of both scatter plots.

The solution of the unconstrained approximative search (Figure 3.4) improves the separation between the mouth and nonmouth from Figure 3.6 significantly, namely to 2.86. The weighted correlation between the mouth and the template then equals 0.53 and between the nonmouth and the template only 0.20.

The weighted regression line and weighted mean computed with the weights obtained by the unconstrained approximative search (Figure 3.4) are shown in Figure 3.8. On the left there are

results for the mouth. The red line is not very different from that in the left part of Figure 3.7. The template has the grey value 0.711 in both of the pixels with the highly influential weights. These are not shown in the scatter plot. The mouth has grey values 0.851 and 0.858 in them, which means that the mouth is almost perfectly symmetric in these two pixels. In the left image of Figure 3.8 these points with the largest weights can be found very close to the intersection of the red and blue lines. Therefore they do not influence the slope of the red line. The intersection corresponds to the weighted mean of the grey values of the template and the weighted mean of the grey values of the mouth.

The scatter plot of the nonmouth against the template is shown in the right part of Figure 3.8. The two pixels of the nonmouth corresponding to the largest weights have the grey values equal to 0.774, so the nonmouth is perfectly symmetric in the two pixels. These two points are almost precisely in the intersection of the red and blue lines in the scatter plot. These do not influence the slope of the red line. The lips do not have so large weights and are not so strong leverage points any more. Moreover the pixels between the nostrils in the nonmouth have relatively large grey values (light pixels). These correspond to low grey values of the lips of the template. This makes the slope lower, in other words pushes the line to be close to horizontal. The red line is then not so different from the blue line of the weighted mean and the weighted correlation is low.

Now we discuss the performance of the solution of the unconstrained approximative search from a more general point of view. The lips are an important feature in the regression of a particular mouth against the template, although they do not have very large weights. For example in the left image of Figure 3.8 they have a leverage effect and determine the slope. In the scatter plot the largest weights correspond to points in the middle of the bulk of the data and the weighted correlation remains sensitive to leverage points just like the sample correlation. The leverage effect of the lips and the (more or less decent) symmetry of every mouth allow it to have a large weighted correlation with the mouth template.

For the nonmouth of the example we can repeat the previous paragraph. The corresponding scatter plot is then the right image of Figure 3.8. The lips have a leverage effect and strongly influence the slope. The nonmouth does not resemble the template in the lips area and the weighted correlation is low. The lips do not need large weights, because they are influential even with smaller weights. However the leverage effect is only moderate rather than strong. The lips namely may not have large weights. Very important lips would influence the regression line very strongly. Points corresponding to the lips would then lie on the line or near it, which would make the error sum of squares for the nonmouth very low and the weighted correlation large.

There is a much larger variability among nonmouths than among mouths. A different situation occurs for nonmouths strongly nonsymmetric in the two highly influential pixels. These push the weighted regression line to be close to horizontal. For any possible intercept and slope the points lie far from the weighted regression line and that makes the error sum of squares large. Then the weighted correlation is low.

Still there is another sort of nonmouths which are symmetric but have nontypical grey values in the highly influential pixels, namely much smaller or larger than in the rest of the nonmouth. That makes again the error sum of squares larger and decreases the weighted correlation.

Although there are these different sorts of nonmouths, each of them is well separated from the mouth thanks to a suitable combination of the effect of the highly influential pixels and the leverage effect of the lips. This explains the existence and the position of the very large weights.

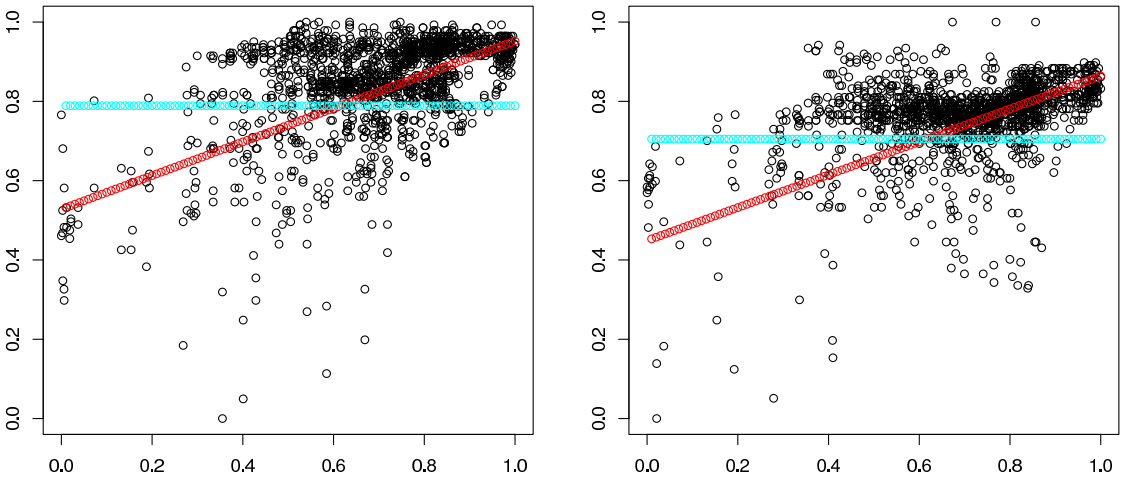

Figure 3.7: Weighted regression and weighted mean with radial weights. Left: mouth against the template. Right: nonmouth against the template.

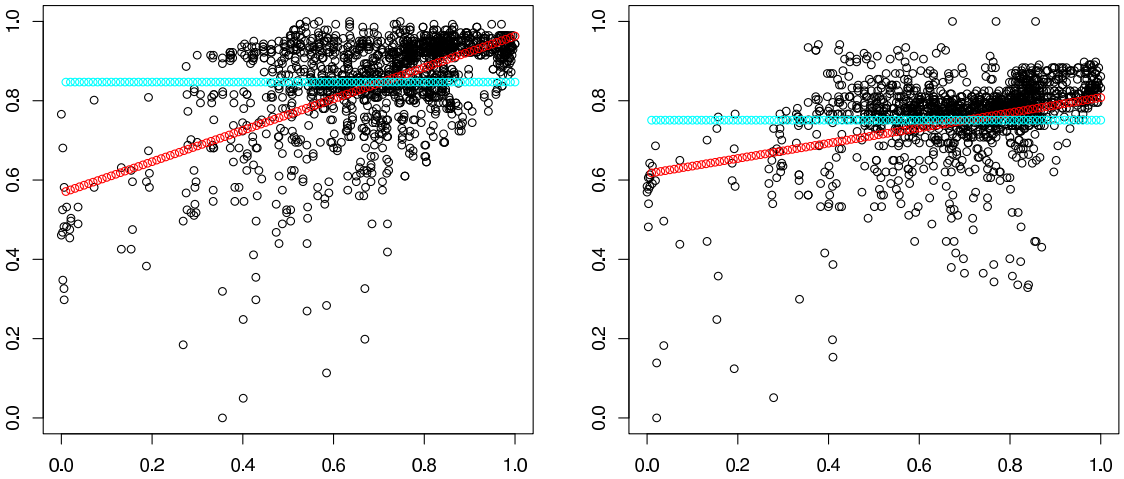

Figure 3.8: Weighted regression and weighted mean with the weights from Figure 3.4. Left: mouth against the template. Right: nonmouth against the template.

We do not see any special property in the highly influential pixels over the whole database of images. These could be shifted by a pixel aside while retaining to have a good separation property. Nor the eigenmouths computed in Chapter 2.5 from all mouths from the database report any special property of the highly influential pixels of Figure 3.4.

Remarks

Because we use the minimax approach, the result is influenced only by several mouths and nonmouths near the boundary between the two classes. Indeed only such nonmouths contribute to the optimization of (3.3) over all weights, which have the weighted correlation with the template relatively large as compared with the mouth. This is a similar situation with support vector machines (see Osuna et al. 1997), which base the classification only on objects (mouths and nonmouths) near the boundary between the classes.

There are usually low or zero weights assigned to the pixels corresponding to the moustache in the solution of both the analytical and approximative searches, which could impugn using the bearded template (Figure 2.2). This was however selected in Chapter 2.3 from a set of seven mouth templates as that with the best ability to locate the mouth correctly in both bearded and nonbearded pictures. Actually the bearded template is a compromise between bearded and nonbearded templates, because the moustache is not so dark and also the chin and cheaks are without any beard.

We have constructed artificial weights from those in Figure 3.4 by removing the two influential pixels with the largest weights 0.13 and placing these large weights to the moustache. The mouth is then located correctly in 98 % of images. The problems appear in nonbearded images.

In general the moustache area has a very large diversity among different people and cannot have large weights for any choice of the template. We have observed that the optimization of weights with a different template without a moustache (Figure 2.8 on the left) finds bearded people as worst cases and again low or zero weights correspond to the moustache area with such template.

The highly influential pixels serve as a protection against extremely nonsymmetric nonmouths, but at the time turn out to be robust to slight nonsymmetry. This will be documented later in Chapter 3.8, where the robustness of different choices of weights to nonsymmetry is examined.

In the optimal weights without constraints there is about one half of the mass of the weights concentrated in only eight pixels, which are shown in Figure 3.4. To examine the importance of these eight particular pixels we have performed the following experiments. First we have reduced the weights in these eight pixels to zero. Then the mouth was located correctly in all 124 images and the worst separation equals 1.73. Next we have started again with the weights from Figure 3.4, kept the weights only in the eight pixels and reduced the weights in all remaining pixels to zero. The mouth was then not found correctly in any image. This shows that also the pixels with the low weights are very important for the weighted correlation. We recall that they contain about one half of the total mass of the weights.

Since the solution of the unconstrained approximative search contains highly influential pixels, it is interesting to examine its sensitivity to a different size, symmetry and rotation of the face and to local changes of the templates. These results will be presented in Chapter 3.8. Results of the approximative search with equal initial weights will be presented only in Chapter 3.5.

3.4 Approximative Search With Constraints

The solution of the approximative search of Chapter 3.3 turns out to contain highly influential points as a consequence of special properties of the training set of images. To obtain a robust version we bound the influence of individual pixels. The algorithm of Chapter 3.3 will be now supplemented by a set of constraints on the weights.

In general the unknown parameters in the unconstrained optimization belong to a much larger space than in a constrained search. Such optimization is slower and more unstable, with a tendency to detect a local extreme or to rely too strongly on the initial weights and pick up special properties of the training set of images. Later in Chapter 3.8 we compare the sensitivity of various possible weights and the constrained solutions turn out to be more robust and therefore preferable to the solution of the unconstrained search.

Just like before we require the weights to be symmetric. Let the weights after being transformed to a vector be denoted by $\boldsymbol{w} = (w_1, \ldots, w_n)^T$. Moreover we require that

$$0 \leq w_i \leq c \quad \text{for every } i = 1, \ldots, n \tag{3.8}$$

with a certain constant c. We have performed the computation with values $c = 0.02$, $c = 0.01$ and $c = 0.005$.

The approximative algorithm of Chapter 3.3 can be used, which modifies weights in pixels selected at random. The weights in the selected pair of pixels can be increased if the new value does not exceed the bound c. At some point during the computation there appear pixels, in which changing the weights would improve (increase) the value of (3.3), but the constraints (3.8) would not be fulfilled any more. The weights in these should be set to c as required in (3.8). Later during the algorithm the weights actually decrease, because the weights are changed in other pixels and that decreases all other weights as we transform the weights to fulfill $\sum_{i=1}^{n} w_i = 1$. Therefore we remember all these influential pixels and assign them the correct value c each time after some of the weights are changed and their new sum is computed.

During the algorithm there is always one particular pixel selected and let its weight be denoted by w^*. This weight in this pixel and also in its symmetry counterpart should be increased by a certain fixed constant δ, for example $\delta = 0.001$. This can be done if the condition

$$\frac{w^* + \delta}{1 + 2\delta} \leq c$$

is fulfilled. This is the condition on the new weights not to exceed the upper bound c. Solving this inequality gives the condition

$$w^* \leq c(1 + 2\delta) - \delta. \tag{3.9}$$

If this is fulfilled, the two pixels obtain weights $w^* + \delta$. Then the sum of the weights is not 1 and the weights must be standardized. If (3.9) is not fulfilled, the weights reach the upper bound or are only slightly lower. Then we store the coordinates of the two pixels and these will obtain exactly the upper bound. Also in this case the following standardization must be carried out.

Let us say there are b pixels which should have their weights equal to the upper bound c. Let the sum of the weights in the remaining pixels equal s. These pixels will be multiplied by a constant α so that the sum of the weights becomes 1. The condition is therefore $bc + \alpha s = 1$. It follows to multiply such pixels, which should not have the weight equal to the upper bound, by

$$\alpha = \frac{1 - bc}{s}.$$

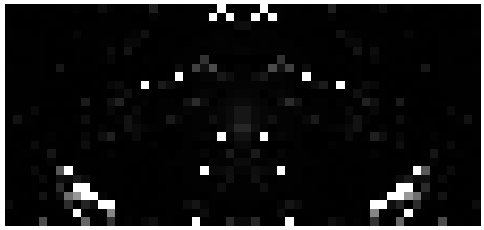

Figure 3.9: Solution of the constrained approximative search with $c = 0.02$.

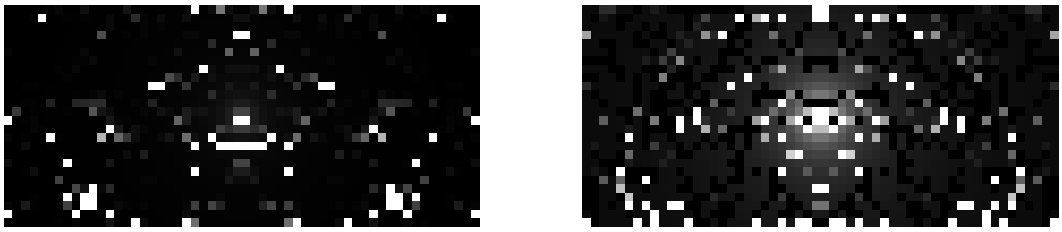

Figure 3.10: Solution of the constrained approximative search with different values of the upper bound c. Left: $c = 0.01$. Right: $c = 0.005$.

It can happen that there is a total amount of $1/c$ weights equal to c so that no other weights can be positive and none of the zero weights can be increased. In that case a possible solution is to allow the weights to decrease rather than requiring that they must attain exactly the value c.

These weights are used to search for the mouth in all images and it is checked if (3.3) increases. In that case we retain the new weights and repeat the process, first trying to increase the weights further in the same two pixels. If the overall separation does not decrease, we keep the weights (before increasing in the two pixels) and select new two pixels. If it is not possible to increase the weights even once, we try to decrease the weights. Then the weight in the two selected pixels is reduced to $\max\{0, w^* - \delta\}$ and the standardization is performed to bring the sum of the weights to 1.

Finally we also try to decrease the weights which attain the upper bound c.

Results of the approximative search with constraints

We now describe the results of the approximative search with constraints (3.8) with different values of c, namely $c = 0.02$, $c = 0.01$ and $c = 0.005$. All examples start with radial initial weights.

We have computed the weights with $c = 0.02$ twenty times. Although the results were not identical, the very lowest separation over all images (3.3) was found twice, up to the precision of 10^{-15}. These best weights are shown in Figure 3.9. The worst separation equals 1.97. This is slightly worse than in the unconstrained case, where the worst separation was 2.10. Still the result is considerably better that with radial weights, for which the worst separation equals only 1.11.

There are 28 pixels (white in Figure 3.9), which have the largest weights equal to 0.02. These are situated more or less on similar places as in the case without constraints: in the bottom part of the rectangle or at the boundary of the lips, between the lips and the moustache or in the upper part corresponding possibly to nostrils.

When the best solution with $c = 0.02$ is used as weights to locate the mouth together with the bearded template, the worst image over the whole training set is a nonbearded man and the worst nonmouth is an area below his mouth. This area contains the lips in its upper part. The midpoint of this area is only 8 pixels below the midpoint of the mouth. This is already considered to be a nonmouth (see Chapter 3.1). The weighted correlation between the mouth and the template equals 0.72, while between the nonmouth and the template 0.43. Their ratio of these values after applying the Fisher's transform equals 1.97.

The others of the twenty solutions of the approximative algorithm with $c = 0.02$ have the largest weights concentrated in the bottom part and many of them are therefore visually similar to Figure 3.9. The other larger weights make a vertical pattern in the right and left side of the rectangle, aside the lips and moustache. In each of the results the upper bound $c = 0.02$ is attained in about 20 or 30 pixels.

From the twenty different solutions of the approximative algorithm with $c = 0.02$ we have computed the mean. The largest weights are equal to about 0.0075 and are obtained in four pixels near the bottom corners, similarly with Figure 3.4 (left). Other larger weights not exceeding 0.005 are assigned to pixels in an oval near both sides of the template and the bottom of the lips. The worst separation of such mean weights then attains however only 1.61, which is much less than the best of the twenty original solutions. The explanation is that the largest weights do not now exceed 0.0075 and do not reach the upper bound $c = 0.02$. Out of the 20 individual results there are 14 better cases than 1.61 and in 6 cases the worst separation is lower. The mean of individuals results of the optimization therefore yields rather weaker results.

For the mean weights we give a list of the worst nonmouths from each of the 124 images. The worst nonmouth is most often situated in the hair, namely in 42 % of images. In 27 % the worst nonmouth is located in the nose, in 20 % in the shirt or sweater. The other areas such as the chin, one of the eyes, eyebrows, neck, moustache or the area below the mouth contain the best nonmouth in not more than 2 % of images.

The solution with $c = 0.01$ is shown in Figure 3.10 (left). Its structure is similar to the solution obtained with $c = 0.02$. There are 56 pixels in which the maximal value $c = 0.01$ is attained. These are located again in the bottom part of the rectangle at the boundary between the lighter and darker parts of the face or next to the lips and moustache, but never in the lips and moustache themselves. On the other hand 36 % of the weights equal zero. These can be found in the moustache and near edges of the rectangle. The separation over all images equals 1.80.

The result with $c = 0.005$ is shown in Figure 3.10 (right). The worst separation over all images (3.3) equals this time 1.44, which is still a considerable improvement compared with the radial weights.

There is a zero weight in 28 % of pixels, which are scattered more or less uniformly over the rectangle. The largest value 0.005 is attained in 72 pixels, which make a pattern of an ellipse circumscribing the mouth. There are many of them in the very bottom and upper part, but also near the midpoint of the mouth. Comparing now Figures 3.9 and 3.10, the pixels with the largest weight c have a similar pattern for different values of c and can be found typically in the lips and in an ellipse around them.

In general the approximative search modifies the weights more quickly in the sense that relatively many weights are reduced to zero and other weights are increased to the upper bound. The constraints play an important role in the computation, they influence strongly the result and prevent the solution from degenerating.

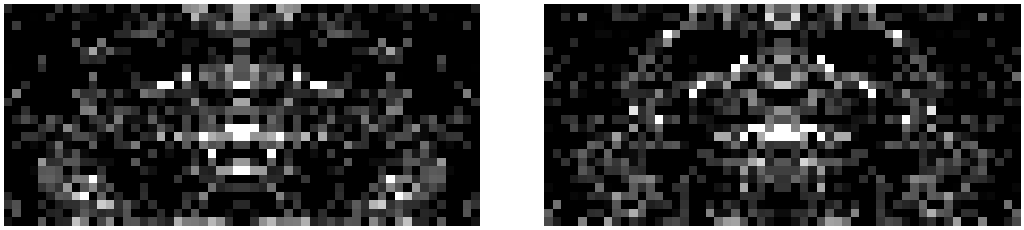

Figure 3.11: Approximative search with $c = 0.005$ modifying 8 pixels (left) and 16 pixels (right) at the same time.

Other approaches

The approximative search modifies the weights in two pixels, namely in one pixel selected at random and in its symmetry counterpart. As another approach we have been modifying a larger number of pixels in the following way. At first there is one pixel selected at random and we consider it together with its symmetry counterpart. With a probability 0.5 we try to increase the weights; we add 0.001 to the weights in these two pixels or set the weights to 0.005, if such new weights would exceed this bound. With a probability 0.5 we try to decrease the weights; we subtract 0.001 from the weights in the two pixels or set the weights to 0, if such new weights would be negative. Now the weights are standardized so that their sum is equal again to 1. The same steps are repeated with a new selection of pixels. In this way we modify 8 or 16 pixels at the same time.

Figure 3.11 shows the results of applying this approximative search on radial initial weights. The upper bound $c = 0.005$ is required in both cases. On the left there are weights modified in 8 pixels at the same time, which means 4 pixels are selected at random and their counterparts are also considered in each step. There the worst separation equals 1.94, which is a big improvement compared to 1.44 obtained with the simple approximative constrained search with $c = 0.005$ (Figure 3.10 right). With modifying 16 pixels in each step we have obtained the right image of Figure 3.11 and the worst separation 1.89, slightly lower than with modifying 8 pixels. The two solutions of Figure 3.11 are similar to each other. The large weights equal to 0.005 are present in the lips near the midpoint, then above the lips under the moustache and some also near the bottom corners, similarly with weights from Figure 3.10. The upper bound 0.005 is attained in 2 % of pixels. On the other hand 49 % of weights are equal to 0 in the when the method modifying 8 pixels is applied and 51 % are equal to 0 with modifying 16 pixels in each step.

The improvement of the separation is dramatic. The constrained approximative search modifying more pixels is slower compared to the simple approach in which the weights are modified only in two pixels. It would be computationally infeasible to examine all possibilities in selecting 8 pixels at random. Each pixel is selected from $26 \times 28 = 728$ possibilities and there are about 10^{18} possible choices of selecting 8 pixels out of 728 pixels. Therefore we let the algorithm run for a fixed period of time, for example 24 hours, although a further running could bring further improvements.

It is possible to run the constrained approximative search modifying 2 pixels and then to apply the approach modifying a larger number of pixels. However then we have observed almost no improvements in the worst separation.

We have also formulated the constraints in a different way. Denoting the radial weights by $\tilde{w}_1, \ldots, \tilde{w}_n$, we now require the solution of the approximative search $w_1, \ldots, w_n$ to fulfill the

condition

$$0 \leq w_i \leq 5\tilde{w}_i \quad \text{for every } i = 1, \ldots, n.$$

This does not allow the solution to differ too much from the initial radial weights. The separation (3.3) with such optimal weights equals 1.47. Maximal weights of about 0.010 appear at the bottom edge of the bottom lip. The midpoint itself has the weight 0.008. In general the middle area of the template obtains the largest weights. Concerning the remaining areas, there are larger weights again in an ellipse similarly with the right image of Figure 3.4. In 31 % of pixels the weight equals zero exactly.

A faster analogy of the approach adding 0.001 to the weights in two pixels is for example to multiply the weights by 2. Then the algorithm starts by checking a condition analogous to (3.9). This is now formulated as

$$\frac{2w^*}{1 + 2w^*} \leq c,$$

which corresponds to the condition

$$w^* \leq \frac{c}{2(1-c)}.$$

If this is fulfilled, the weights in the selected two pixels are multiplied by 2 and the standardization of the weights is performed. Otherwise the weights are either multiplied by one half or set directly to zero if they are already small enough. The standardization of the weights is then necessary. This approach turns out not to improve the worst separation as much as the approach adding 0.001 to the weights.

Another possibility is to try to add any of the constants $0.001, 0.002, \ldots, 0.005$ to the weights in the two pixels. This is more or less a discretized version of examining all possible weights with a large computational intensity.

3.5 Two-Stage Search

In this chapter we recommend the two-stage search to approximate to the solution of maximizing (3.3) over the weights. It combines two algorithms described above. Moreover we present results obtained with different initial weights. All the weights of this chapter are required not to exceed the upper bound $c = 0.005$.

There are two approaches to approximating the optimal weights defined by maximizing (3.3) over all possible weights. One is the analytical search described in Chapter 3.2; for the approximative search we prefer the constrained version of Chapter 3.4. The solutions of these two approaches have a very different appearance. The following two-stage search is a combination of both methods. We start with some initial weights and apply the analytical search. The approximative search can be further applied on its output to set out from the local optimum.

While the analytical search handles whole areas of pixels and changes the weights slightly, the approximative approach handles pixels more individually than the analytical one and allows the weights to be changed more substantially. Their combination improves the separation in the worst case and we use at the same time a reasonably small upper bound (c=0.005) to prevents the weights from degenerating. We use again the bearded template (Figure 2.2) and the database with 124 images.

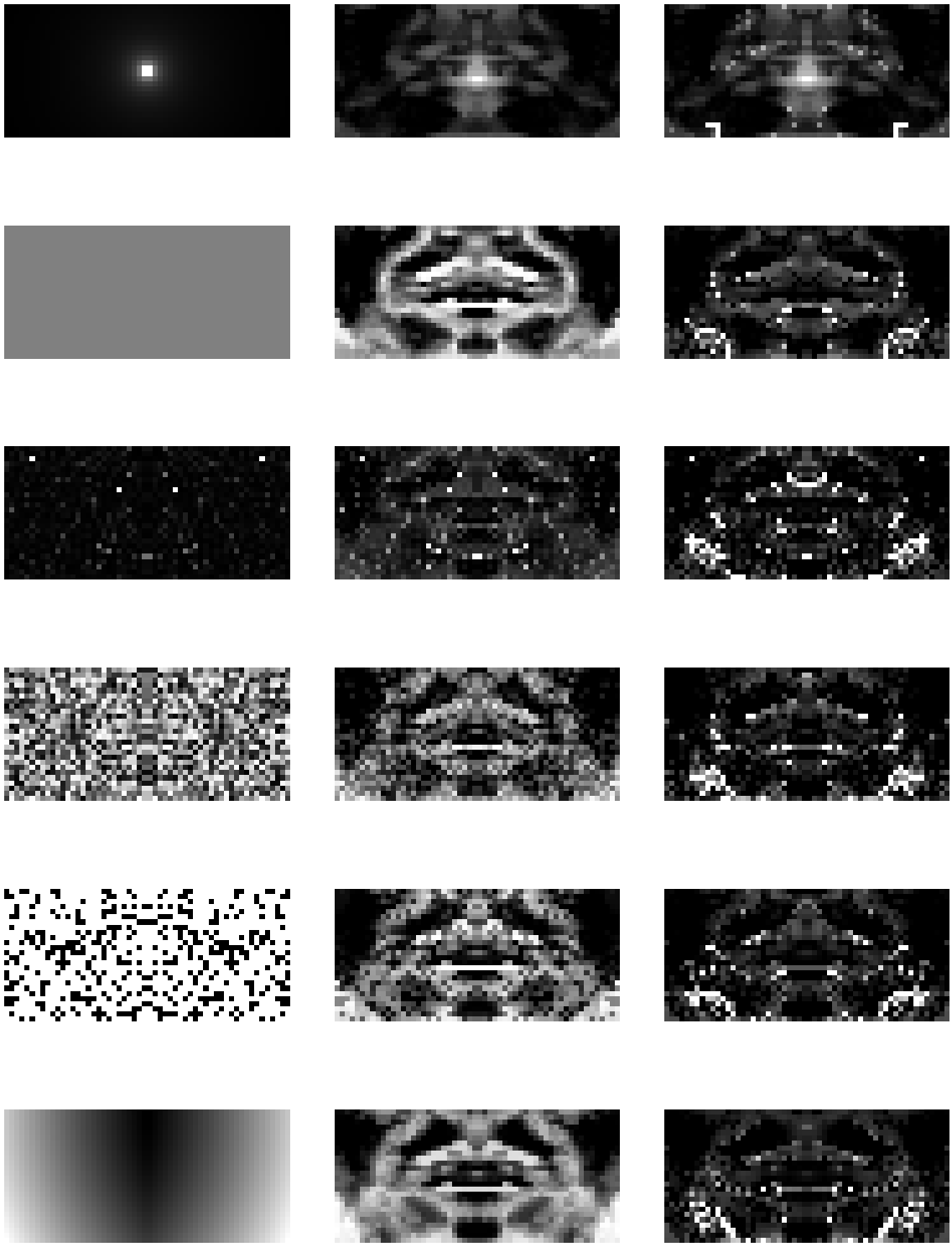

Figure 3.12: Results of the two-stage search with different initial weights. In each row: initial weights (left), result of analytical (middle) and two-stage search (right) for optimal weights with the bearded template.

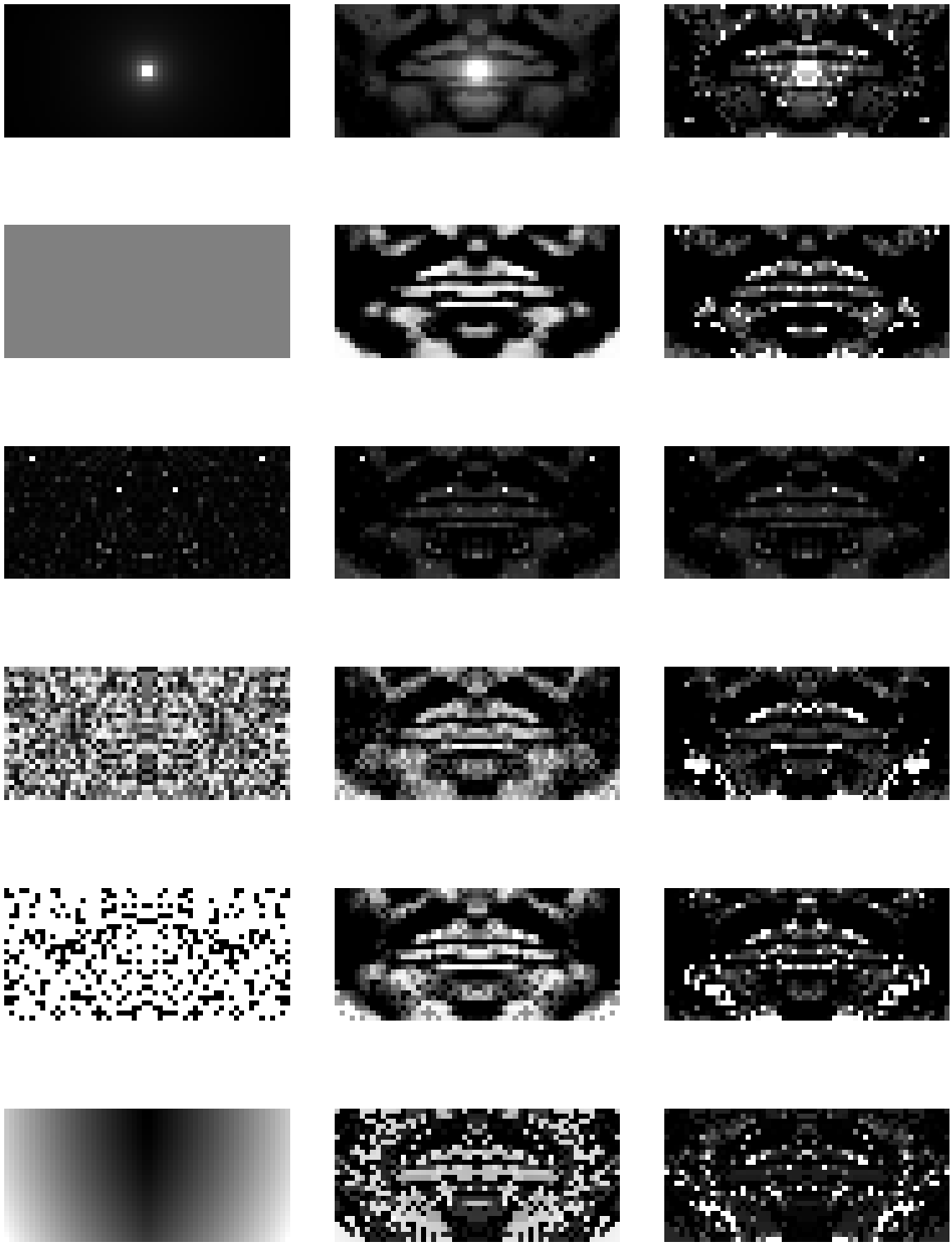

Figure 3.13: Results of the two-stage search minimizing the difference between the mouth and nonmouth. In each row: initial weights (left), result of analytical (middle) and two-stage search (right) for optimal weights with the bearded template.

Let us start with radial weights, because they perform better in locating the mouth than equal ones. The solution of the analytical search has already been presented. The result is repeated in Figure 3.12, which gives a systematic overview of results for different initial weights. In the first row the radial weights are shown again, then the result of the analytical search and finally on the right the result of the two-stage search, which is obtained by applying the constrained approximative search on the solution of the analytical search shown in the middle of the row.

We recall that the initial worst separation with radial weights equals 1.11. The analytical search (after modifying the radial weights not to exceed 0.005 in any pixel) improves this worst value to 1.68. We apply the approximative search on these weights. The result is shown in the right image of the first row of Figure 3.12. The worst separation now attains 1.73, which is not a big improvement. Larger weights correspond to some of the bottom pixels of the template and also to the midpoint of the template. The upper bound 0.005 is attained in 12 pixels. On the other hand a zero weight is attained in 19 % of pixels. These are black pixels in the image and are situated in the top corners and in the bottom part, between the lips and the white pixels with the largest weights.

It is possible to apply the analytical search again on the solution obtained so far. Improvements in such situations are however only marginal.

Now we can compare the results obtained with the two-stage search with different initial weights. With equal weights the mouth is not located in all pictures, but only in 94 % of them. The worst separation is therefore less than 1, namely 0.78. The solution of the analytical search improves this value to 1.67 and is shown in the middle image of the second row of Figure 3.12. Larger weights can now be found in parts of the lips and then in the bottom part of the rectangle. The largest weights however do not exceed 0.002, although the upper bound $c = 0.005$ was allowed to be attained. There is a zero weight in 20 % of the pixels. The mouth is located correctly in all images and with a separation comparable with the solution obtained with radial initial weights. A further big improvement is obtained by applying the approximative search. The solution shown in the right image of the second row of Figure 3.12 gives the worst separation 1.94. The upper bound 0.005 is attained in 2 % of pixels and there is a zero weight in 38 % of pixels.

Comparing the result of the two-stage search with radial initial weights with that with equal initial weights, their structure is different. The two solutions with different initial weights are shown in right images of the first and second row of in Figure 3.12. A common feature of both solutions is assigning larger weights to some pixels in the bottom part of the template, rather than stressing entirely the lips. It turns out that the solution depends heavily on the initial weights. Moreover better result is here obtained with worse initial weights. Therefore it can be recommended for a general situation to run the optimization several times with different initial weights and compare the results.

The mean of these two sets of weights attains the worst separation of only 1.47. Also other possible convex linear combinations of the two sets of weights do not improve the worst separation.

The worst separation for various weights is compared in Table 3.2. The left half uses the approach described above, while the results of the right half use a different separation between the mouth and nonmouth, which will be described later. In each case the bearded template is applied to search for the mouth. We still continue using the separation function (3.1) and we have already described the first two rows devoted to radial and equal initial weights. Particular rows of Table 3.2 correspond to rows of images in Figure 3.12. The left column of the left

Table 3.2: Performance of different weights of Figure 3.12 in locating the mouth with the bearded template. Left: the separation (3.1), right: the separation (3.10) is used.

	Worst separation: ratio			Worst separation: difference		
Weights	Initial	Analytical	Two-stage	Initial	Analytical	Two-stage
Radial	1.11	1.68	1.73	0.08	0.29	0.40
Equal	0.78	1.67	1.94	−0.11	0.23	0.38
3.	0.77	1.45	1.81	−0.04	0.23	0.23
4.	0.80	1.52	1.97	−0.10	0.23	0.40
5.	0.82	1.54	1.85	−0.09	0.23	0.44
6.	0.91	1.33	1.88	−0.11	0.24	0.31

half of the table contains the worst separation with the initial weights from the left column of Figure 3.12, the middle column of the table contains results of the weights obtained by the analytical search and the right column corresponds to the solution of the two-stage search, just like in Figure 3.12.

The idea of the following four choices initial weights is to examine other possibilities with a structure which may seem unusual or unreasonable or completely random. All of them are symmetric. The optimization procedures were applied always under the requirement that they do not exceed the upper bound $c = 0.005$.

The third set of weights is a random permutation of radial weights. The pixels with the largest weights 0.005 are situated near the top corners and above the lips below the moustache. The fourth set of initial weights are generated from a uniform distribution. In order for the sum of the weights to be 1, the weights must be smaller than about 0.0014. In the fifth set of initial weights we have randomly selected 25 % of the pixels. These have a zero weight and the weights of the remaining pixels are equal, so the value of each weight is equal to about 0.0009. The weights in the last row of Figure 3.12 (and at the same time of Table 3.2) have the smallest weights in the top middle part and increase as the distance from it increases. Therefore the largest weights are in the bottom corners.

The solutions of the analytical search copy to some extent the structure of the initial weights. This is remarkable in the third row, where the large weights remain in the influential pixels of the initial weights. Also in the fourth and fifth row there are very different weights in neighbouring pixels similarly with the initial weights. Only the solutions of the two-stage search seem to have a structure not so dependent on the initial weights, namely pixels with large weights are present in the lips and in the bottom part near the corners.

The performance of random weights is weaker than that of radial weights. Radial initial weights lead also to the best solution of the analytical search. However a further application of the approximative search brings a dramatic improvement of the worst separation for some of the random weights. The best result is obtained with the solution of the two-stage search shown in the fourth row of Figure 3.12. The solution with equal initial weights is only slightly worse. For practical use it can be recommended to start with different initial weights or to use equal weights.

Finally let us compare the analytical and approximative search. The analytical search changes the weights only by very small constants. Starting with radial weights, the largest weights are retained in the midpoint. Starting with equal weights, which are much lower than

0.005, this upper bound is not attained at all. It is however attained by the two-stage search easily. The separation with the bound 0.005 is attained when equal weights are iteratively optimized using the two-stage approach. Then almost 38 % of the pixels have the weights exactly zero, most of which correspond to the top corners of the template. In other words only a part of the 1456 pixels of the template is actually used.

Optimizing the Difference as the Separation Measure

A different possibility is to consider the separation measure as a difference instead of the ratio in (3.1). Let us denote again the vector of grey values of a particular mouth by $\boldsymbol{x} = (x_1, \ldots, x_n)^T$, nonmouth by $\boldsymbol{z} = (z_1, \ldots, z_n)^T$, template by $\boldsymbol{t} = (t_1, \ldots, t_n)^T$ and given weights by $\boldsymbol{w} = (w_1, \ldots, w_n)^T$. The separation formulated as

$$f(\boldsymbol{x}, \boldsymbol{z}, \boldsymbol{t}, \boldsymbol{w}) = r^F_W(\boldsymbol{x}, \boldsymbol{t}; \boldsymbol{w}) - r^F_W(\boldsymbol{z}, \boldsymbol{t}; \boldsymbol{w}). \tag{3.10}$$

is not equivalent with the ratio (3.1).

Let us recall that the Fisher's transformation brings the distribution of the classical sample correlation coefficient close to normal and it is approximately true that its variance depends only on the number of observations, while an analogous result for the weighted correlation is unknown. It is reasonable to consider the difference instead of the ratio of two variables with a normal distribution, because its interpretation is straightforward and also the difference is standardly used for example to test the null hypothesis of equality of two population correlation coefficients.

We have repeated the computations of the analytical and two-stage search with the bearded template with all initial weights from Figure 3.12. These are repeated in the left column of Figure 3.13 and the results obtained by the analytical search with (3.10) are shown in the middle column. These have been further used as starting weights for the approximative search and the resulting weights are given in the right column of Figure 3.13. In each computation the upper bound c=0.005 is required.

The worst separation for each of these weights is shown in the right half of Table 3.2. Positive separation (3.10) means that the mouth is correctly located in each image. For example using the radial weights, the worst mouth has the weighted correlation with bearded template 0.61. The worst nonmouth is the middle of the nose of the same person with the weighted correlation with the template 0.56. The worst separation with radial weights is the difference of these two values after performing the Fisher's transform on both of them, namely 0.08. This value can be found in the first row of Table 3.2. The solution of the two-stage search with radial initial weights is shown in the top right corner of Figure 3.13. The worst mouth with these weights has the weighted correlation with the template 0.78. The worst nonmouth is the area below the bottom lip with the weighted correlation with the template 0.55. The worst separation is then equal to 0.40.

The best results with the separation (3.10) are obtained with the random weights from the fifth row of Table 3.13, which are equal weights with the exception of 25 % of pixels, where the weight is zero. These initial weights yield the worst separation -0.09, namely the mouth has the weighted correlation with the template 0.40 and an area in the hair 0.47, so the separation (3.10) between them is negative. The solution of the two-stage search is shown in the fifth row in the right column of Figure 3.13. The worst image is now different from the worst image with the initial weights and the worst mouth has now the weighted correlation with the template 0.76 and an area near the top boundary of the hair of the person 0.50, so the worst separation equals 0.44.

Now we can compare the separation defined by the ratio (3.1) and the other approach defined by the difference (3.10). Resulting weights of Figure 3.12 are not identical with those from Figure 3.13. However we can observe that the important weights always correspond to the bottom areas below the mouth and to certain pixels of the lips. A big difference between the two methods is obtained only in the third row, which yields only poor results.

There is a general tendency that both the mouth and nonmouth using the ratio (3.1) have a smaller weighted correlation with the bearded template than using the difference (3.10). The mouths in the whole database have the sample correlation with the template between 0.35 and 0.70 and the worst nonmouth in every image has the correlation with the template between 0.35 and 0.59. When the weights are used obtained from equal initial weights and optimizing the ratio by the two-stage search, the weighted correlation of the mouths with the template increases to 0.45 to 0.80 and it decreases for the worst nonmouths to 0.22 to 0.39. These results are different when the weights obtained by optimizing the difference rather than the ratio are used. The mouths then have a larger weighted correlation between 0.68 and 0.93, while the worst nonmouths in particular images have also increased values between 0.37 and 0.65.

In general we can say that the ratio as the separation measure increases the weighted correlation between the mouth and the template and decreases the value between the nonmouth and the template. On the other hand the difference as the separation measure increases both of these weighted correlations. As a different example let us mention radial weights. The worst nonmouths in different images have the weighted correlation with the template between 0.39 and 0.61, when radial weights are used. These values decrease to the interval between 0.25 and 0.45 with weights obtained by optimizing the ratio, but decrease to 0.46 to 0.67 with weights obtained by optimizing the difference. Both optimizations were carried out using the two-stage search. It is desirable that the nonmouth becomes less correlated with the template and this gives an argument in favour of the ratio as the separation measure.

Moreover we have observed that as the ratio is optimized, also the difference between the worst mouth and nonmouth is improved compared to the difference obtained with initial weights. The same tendency is true for the other case, namely the ratio of the worst mouth and nonmouth improves when the difference is optimized. This shows that there is no contradiction between the two separation measures, although they are not equivalent and approach the problem from a different point of view.

There is no unambigous answer to the question, which of the definitions to the separation should be used. The interpretation of the difference is more intuitive than that of the ratio, but for the reasons mentioned above we prefer the ratio (3.1) and this will be used as the separation measure also in the following chapters.

Finally we recall that we have recommended to use the equal initial weights for the separation in the form of the ratio (3.1). Equal weights are however rather weaker for the separation in the form of the difference (3.10), for which random weights are more suitable. Rather than to recommend the difference together with random or unusual weights we believe that the ratio together with equal weights is a reasonable choice.

Other Approaches

We have also combined the two approaches of Chapters 3.2 and 3.4 in the other way, that means first the approximative search and then the analytical one. Starting with radial weights, the solution of the approximative search with $c = 0.005$ has been shown in Figure 3.10 (right) and we recall that the worst separation equals 1.44. When the analytical search is applied to

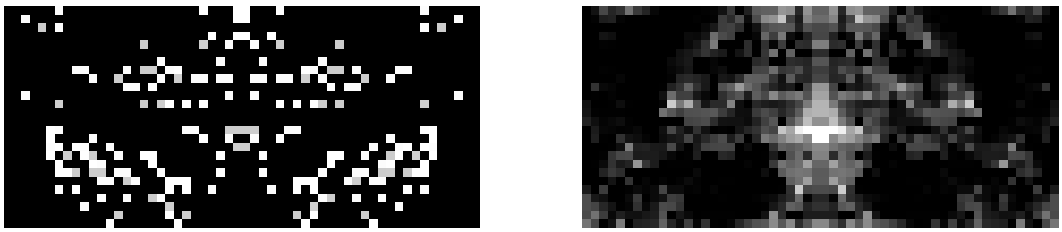

Figure 3.14: Left: result of the two-stage search with the mouth teplate of Figure 2.8 (left) and radial initial weights. Right: result of the modified two-stage search with the bearded template starting with radial weights; the approximative search was used modifying the weights in 8 pixels at the same time.

optimize further these weights, there is no visible difference of the result from the right image of Figure 3.10. The worst separation is however improved to 1.73. The difference between the two sets of weights is apparent only in the log scale, because there are small changes of the weights. The approximative search namely sets some pixels to zero, while others are retained without change, and makes weights in neighbouring pixels very different. The analytical search keeps the large weights, but otherwise the small weights all over the template become smoother. Some areas have the weights reduced to zero just like in the left image of Figure 3.2, but weights in other areas become very small positive. Therefore the solution can be interpreted as the left image of Figure 3.2 supplemented by the white pixels with large weights of the right image of Figure 3.10.

However it would not be reasonable to start with the approximative search. This namely modifies the weights with a brutal force and can otherwise lead the optimization in a wrong direction. The analytical solution is available and it should be used to optimize the overall criterion as far as possible and when this way is exploited we can try the approximative search.

We have computed the optimization also with a different template. This is not necessary, because we find the mouth correctly in the whole database using the bearded template. However to see a comparison we use the mouth from the person in Figure 1.1 as a template. This is shown in the left part of Figure 2.8 in the size 26×56 pixels. This template together with radial weights locate the mouth correctly in 97 % of images and the worst separation equals 0.81. Then we apply the analytical search and require the weights to be smaller than the upper bound $c = 0.005$. The separation increases to 1.32. The solution still has the largest weights in the middle area and looks very similar to the left image of Figure 3.2. There are only two pixels in the midpoint with the largest weight 0.005. On the other hand there are 2 % of pixels on the left and right sides of the rectangle with the weight exactly zero.

The approximative search was applied on the the solution of the analytical search. The worst separation is further improved to 1.55. The resulting weights are shown in Figure 3.14 (left). Out of the 1456 pixels, 86 % have the weight equal to zero, 11 % to the upper bound 0.005 and only 3 % attain a positive value lower than the upper bound.

The mouth in the left part of Figure 2.8 was simply selected as the mouth of the person from Figure 1.1, we have not performed any transformation on it and it is still able to locate the mouth successfully in the database with 124 images when suitable weights are found. On the other hand the bearded template was obtained in a more complicated way. This is the mean of four bearded templates (as described in Chapter 2.3) and it turned out to be the best in a set

of seven different templates. The worst separation obtained with the bearded template was 1.73 in the same situation with the two-stage search. So the complicated search for the template brings now benefits, compared to a simple taking one raw mouth to be the template.

In Chapter 3.4 we have described the approximative search with modifying weights in a larger number of pixels at the same time. We have applied this approximative search also on the solution of the analytical search. Starting with radial weights the solution of the analytical search has been shown in Figure 3.2 and the worst separation equals 1.68. Starting with these weights, the approximative search modifying 2 pixels at the same time has been presented in Chapter 3.4, the solution is shown in the top right corner of Figure 3.12 and the worst separation equals 1.73. Starting with the solution of the analytical search and applying the approximative search modifying 8 pixels at the same time improves the worst separation to 1.79. The weights are shown in the right part of Figure 3.14. The version modifying 16 pixels at the same time however does not bring any further improvement beyond 1.68. One reason can be that the algorithm would need an infeasible amount of time in order to go over all possible choices of 8 pixels. Another problem is the probability of increasing or decreasing weights. The weight is increased or decreased according to a tossing a coin, each possiblity has a probability 0.5 and this may not be optimal.

Also applying this approximative search on results of the two-stage search, in which the approximative algorithm was modifying weights in two pixels at the time, does not bring improvements.

Finally we comment the choice of the upper bound of 0.005 for the analytical search. Only the radial weights and the initial weights in the fourth row of Figure 3.12 attain the value 0.005. The solutions of the analytical search attain this upper bound also only with these two initial sets of weights. In other words increasing the upper bound to a larger value influences the result only for the weights in the fourth row of the figure and also for radial weights, for which this effect has been already described in Chapter 3.2.

Starting with the weights from the fourth row of Figure 3.12 we have applied the analytical search with a larger upper bound $c = 0.02$. The solution looks exactly like the initial weights and the largest weights are equal only to 0.011. The worst separation equals 1.30, while the solution with a stricter upper bound 0.005 yields 1.33. In Chapter 3.2 there was a similar controversial situation with radial initial weights. The problem is in the high dimension of the optimization and complexity of the objective function. Clearly larger upper bounds allow the solution to come to a local extreme and remain there instead of a further approximation to the global extreme.

Existing Approaches to Optimization

Press et al. (1992) describe the Nelder-Mead method as one of methods for multidimensional optimization. This method is called downhill simplex method, although it is completely unrelated to the simplex method of linear programming. It does not require computing the derivatives of the objective function and in principle examines the value of the objective function in vertices of different simplexes. In the context of minimization it makes downhill moves in a straightforward fashion, usually moving one vertex to the opposite face of the simplex to obtain a lower value of the objective function or expanding or contracting the simplex.

We believe that such approach is not very suitable, but rather too simple for such a high dimension. Also the storage requirement of the Nelder-Mead method is of order N^2 if N denotes

the dimension of the problem. In our case the computation requires storing a matrix with about a million elements. Moreover our optimization described in previous chapters improves the separation for the worst case, which is not the same throughout the whole computation but we search for the worst case (or several worst cases) each time in each step of the iterative search. Therefore the approximative search searching for the optimal weights over the whole database of images cannot be replaced by existing automatic methods for optimization.

Nevertheless we have performed numerical experiments with the Nelder-Mead method, which is implemented in the software package R in the function `constrOptim()`. We have optimized the separation only between one particular mouth and nonmouth, namely those from Figure 2.5. The separation between them using equal weights is 0.90. We have applied the constrained optimization in R to maximize the separation between the mouth and nonmouth as a function of weights. We require that the weights lie in the interval between 0 and 0.005 and their sum equals to 1. The constraints on the symmetry of the weights are formulated as equalities. Because the function is able to process constraints only in the form of inequalities, we reformulate the conditions so that the difference of the two weights in the two symmetry counterparts does not exceed ε and at the same time exceeds $-\varepsilon$. For a small value of $\varepsilon = 10^{-6}$ the separation is not improved at all. For a larger value of $\varepsilon = 10^{-4}$ the separation is improved only to 1.01, but the weights are apparently not symmetric, because the value of ε is already too large.

On the other hand the optimization without the conditions on the symmetry improves the separation very well to 1.80. There are weights reduced to zero in the top corners of the template corresponding to the corners in the nonmouth (Figure 2.5 right). Similarly with our analytical search of Chapter 3.2, the optimization takes place in very small steps and in the solution there can be larger lighter or darker sections recognized as the method handles pixels as groups rather than individually. Another similar aspect with the experience with our optimization is the strong dependence of the results on the initial solution (initial weights) which must be specified.

While the function `constrOptim()` should be able to optimize under linear inequality constraints, the problem seems to be in the constraints in the form of equalities and the results are not satisfactory even for such a simplified situation with one mouth and nonmouth.

Another existing algorithm for numerical optimization is simulated annealing. This is described by Press et al. (1992) as a relatively new method for combinatorial optimization with promising results in situations with a very complicated objective function containing many local extremes. The basic idea of simulated annealing is applicable also to optimization in a continuous space. The problematic part is however the proposal of a random step from the current system state, in other words creation of a random change in the configuration. The book Press et al. (1992) warns that annealing schemes in a continuous case are typically inefficient, namely propose almost always an uphill move in narrow valleys when a downhill move exists, or become inefficient as convergence to the optimum is approached. Moreover the literature on optimization does not consider simulated annealing to be a standard method yet and there is not yet enough practical evidence with it to say definitely what its future place among optimization methods will be.

3.6 Results in Another Database of Images

In this chapter we work with 88 images of the other database, which also comes from the Institute of Human Genetics. It contains pictures of different people, which are not contained

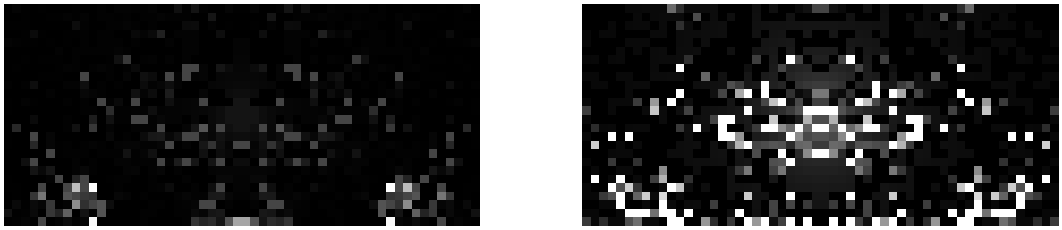

Figure 3.15: Weights obtained by the approximative search over the new database of images starting with radial weights. Unconstrained (left) and constrained search with $c = 0.005$ (right).

Table 3.3: Worst separation for different weights optimized over the new database of images.

Weights	Initial weights	Worst separation
Equal	-	0.55
Radial	-	0.93
Approx. unconstrained	Radial	1.66
Approx. $c = 0.005$	Radial	1.48
Analytical $c = 0.005$	Radial	1.33
Anal. $\Longrightarrow$ approx.	Radial	1.50
Analytical $c = 0.005$	Equal	1.60
Anal. $\Longrightarrow$ approx.	Equal	1.70

in the original database of 124 images. We call these 88 images simply new images and use again the bearded template from Figure 2.2. Neither radial nor equal weights together with the bearded template are able to locate the mouth correctly in every image of the new database.

Radial weights locate the mouth correctly in 99 % of cases and the worst separation equals 0.94. The only problematic image is a picture of an older lady with an unusually big mouth, which is at the same time affected by small rotation, nonsymmetry and a light grimace. The most suspicious nonmouth is a part of the shirt quite similar to that of Figure 2.5. Using radial weights the weighted correlation between the bearded template and the mouth equals 0.42 and between the template and the worst nonmouth 0.46.

Now we apply the analytical and approximative procedure to optimize the weights starting with radial weights. If not stated otherwise, in the following optimization we require the weights not to exceed the upper bound $c = 0.005$. Radial weights transformed to fulfill this condition

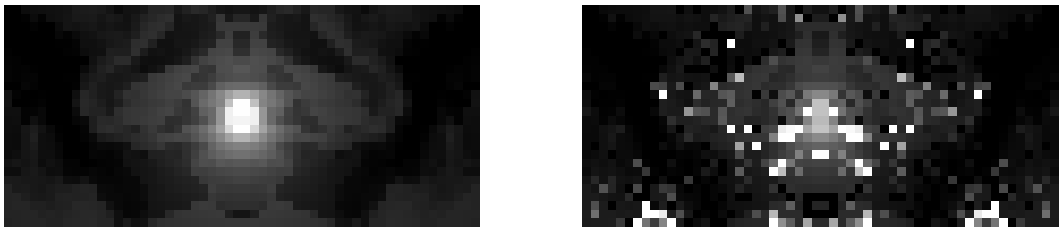

Figure 3.16: Optimal weights in new images. Results with radial initial weights. Left: analytical search. Right: approximative search applied after the analytical search.

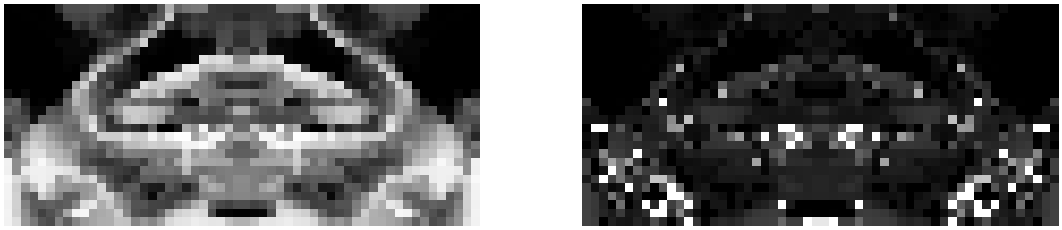

Figure 3.17: Optimal weights in new images. Results with equal initial weights. Left: analytical search. Right: approximative search applied after the analytical search.

locate the mouth again in 99 % of cases with the same problematic image and the worst separation 0.90. Results of different methods optimized over the new database of 88 images are summarized in Table 3.3.

Starting with radial weights the unconstrained approximative search gives the solution shown in the left part of Figure 3.15. The worst separation equals 1.77. The largest weights are equal to 0.019 and can be found in pixels corresponding to the bottom of the template. These are not so much influential, the maximal weight is much lower than that obtained with the original database. On the other hand there are 43 % of pixels which have the weight equal to zero. These can be found scattered everywhere in the rectangle with the exception of the middle area and the bottom corners.

The solution of the approximative search with the constraint $c = 0.005$ starting with radial weights is shown in the right part of Figure 3.15. The separation with these weights equals 1.32. The upper bound is attained in 7 % of pixels situated in the lips and bottom part of the template. As many as 45 % of the weights are equal to zero. These are concentrated in the moustache as well as in the bottom half of the template, mainly below the lips.

The solution of the analytical search with radial initial weights is shown in the left part of Figure 3.16. The worst separation with these weights equals 1.33. The upper bound 0.005 is obtained in 6 pixels in the midpoint. On the other hand 8 % of pixels have the weight equal to zero. These can be found in the top corners and on the right and left side near the edges of the rectangle. Applying the approximative search with $c = 0.005$ on this solution of the analytical search, the worst separation is further improved to 1.50 and the result is shown in the right part of Figure 3.16. Now there are 36 pixels attaining the largest weight 0.005. These correspond to parts of the lips and also to the bottom part of the template, similarly with the weights from Figure 3.10. There are 16 % of weights which are equal to zero exactly. These are concentrated in the top corners and on the right and left sides and others are scattered over the whole rectangle except for the middle part.

Equal weights locate the mouth in 92 % of images of the new database and the worst separation equals only 0.55. The worst image is again the same lady as with the radial weights. The worst nonmouth is now a symmetric area at the top of the ear. There is a wide column of hair with low grey values in the middle and grey values of the face in the left part of the nonmouth are similar to those of the background in the right part. The sample correlation between the mouth and the template equals 0.29 and between the worst nonmouth and the template 0.49. The following paragraphs describe the results with equal initial weights.

The analytical search improves the worst separation to 1.60. The result is shown in the left part of Figure 3.17. The upper bound $c = 0.005$ has been required but not reached and

Table 3.4: Cross-validation of results. The weights are optimized over one database and then applied to the other of the two databases.

		Database			
Weights	Initial weights	Optimize over original	Apply on new	Optimize over new	Apply on original
Equal	-	0.78	0.55	0.55	0.78
Radial	-	1.11	0.93	0.93	1.11
Analytical	Radial	1.68	1.08	1.33	1.26
Anal. $\Longrightarrow$ approx.	Radial	1.73	1.06	1.50	1.26
Analytical	Equal	1.67	1.16	1.60	1.19
Anal. $\Longrightarrow$ approx.	Equal	1.94	1.09	1.70	1.15

the maximal weight equals about 0.0017. These can be found in parts of the lips and near the bottom corners of the rectangle. There are 10 % of weights equal to zero. When this result of the analytical search is used as input for the approximative search with $c = 0.005$, the worst separation is further improved to 1.70 and the solution is shown in the right part of Figure 3.17. The upper bound 0.005 is attained in 46 pixels again near the bottom corners and near the lips. There is a zero weight in 25 % of the pixels.

Comparing results of different methods according to the choice of initial weights, Table 3.3 shows that methods starting with equal weights are preferable to radial weights, although the equal weights themselves perform worse than radial ones. The best solution not exceeding the upper bound $c = 0.005$ is obtained with the two-stage search and the worst separation attains 1.70.

Again no convex linear combination of the weights gives a further improvement of the worst separation. Combining in this way two sets of weights together, the worst separation is then either lower than in any of the two cases or larger than the smaller one but smaller than the larger of the two cases.

The weights have been optimized over one of the two databases of images. We now check the performance of the results always on the other database. The results are summarized in Table 3.4. We use again the bearded template from Figure 2.2. First the results of equal and radial weights are given in the first two rows.

In the left part of the table, the results are optimized over the original database. The table shows the results of the analytical and the two-stage search always with the upper bound $c = 0.005$ with either radial or equal initial weights. These results have been already presented in previous parts of Chapter 3. These results are now applied to locating the mouth in the new database. For example the analytical search starting with radial weights over the original database improves the worst separation to 1.68 and these weights applied to locate the mouth in the new database together with the bearded template locate the mouth again in every image and yield the worst separation 1.08. The performance decreases dramatically compared to the performance in the original database, over which are the weights optimized. The best result is 1.16, attained with the analytical search starting with equal weights. In spite of the decrease in the worst separation each choice of the weights still performs better than equal or radial weights in the new database.

Table 3.5: Worst separation obtained with different weights in the original and new database and over both databases jointly.

Weights	Initial weights	Worst separation Original	New	Both
Equal	-	0.78	0.55	0.55
Radial	-	1.11	0.93	0.93
Analytical	Radial	1.68	1.33	1.33
Anal. $\Longrightarrow$ approx.	Radial	1.73	1.50	1.47
Analytical	Equal	1.67	1.60	1.58
Anal. $\Longrightarrow$ approx.	Equal	1.94	1.70	1.65

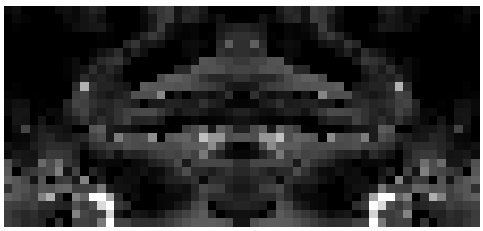

Figure 3.18: The best weights obtained for all 212 images. Starting with equal weights, the analytical and then the constrained approximative search with $c = 0.005$ have been applied.

The right part of Table 3.4 uses weights optimized over the new database. The last but one column repeats the results optimized over the new database itself. These weights are applied to the original database and the results are given in the very last column. There the best of the worst separations attains 1.26. We recall that equal weights have the worst separation 0.78 and radial weights 1.11 in the original database. Again each choice of optimal weights performs better than radial weights, although the difference is either small or negligible.

To summarize the results obtained so far, we have performed the optimization over one of the databases and applied the optimal weights to the other database. The mouth is located correctly in every image in each situation. However none of the choices of weights in Table 3.4 is uniformly better than the others. Moreover applying the approximative search after the analytical one improves the separation in the optimized database but worsens it in the other database.

It is no surprise that the performance of optimal weights decreases, when they are applied to different images. However such drastic decrease of separation between mouths and nonmouths shows that the optimization procedures pick up too special properties of the data or are too strongly influenced by atypical cases. This reveals the high dimensionality of the mouths.

Finally we optimize the weights over the two databases together over the total number of $124 + 88 = 212$ images. The results are presented in Table 3.5 together with a comparison of results separately for the original and the new database.

The bearded template and the radial weights locate the mouth correctly in all images but one. We recall that this problematic image comes from the new database. The worst separation equals 0.93. We have applied the analytical search with $c = 0.005$ starting with radial weights.

The resulting weights are very similar to those in the left image in Figure 3.16, which is the solution of the analytical search over the new images. The middle part remains to have larger weights, which is inherited from the initial weights. However only 3 % of pixels have the weight equal to zero. The worst separation equals 1.33, just like in the database of new images.

We have applied the approximative search with the upper bound $c = 0.005$ on the solution of the analytical search. This time the separation increases to 1.47, which is slightly less than in the new database. The solution is a true copy of the left image of Figure 3.12. Then 20 pixels attain the largest weight and 24 % of the weights are equal to zero.

The equal weights locate the mouth correctly in 92 % of cases, namely there are 8 problematic pictures in the original database and also 8 in the new one. We now apply the same procedures of above with equal initial weights. The solution of the analytical search is similar to that in the left part of Figure 3.16 which was optimized over the new images only. Also the worst separation is only slightly lower than in the new images and atttains namely 1.58. The largest weight equals about 0.0020, so the upper bound $c = 0.005$ is again not attained. There are 13 % of pixels with a zero weight.

The approximative search with $c = 0.005$ was applied on this solution of the analytical search. The worst separation increases further to 1.65 and the resulting weights are shown in Figure 3.18. These are very similar to the right image of Figure 3.12, in other words to the solution of the combined search over the original database, which starts also with equal initial weights. There are now 8 pixels with the weight equal to the upper bound. These are situated in the bottom near the corners and further there are 22 % of pixels with a zero weight.

We can conclude that results optimized over the two databases joined together are similar to results optimized over the new database, which is true for the visual appearance of the resulting weights as well as for the worst separation. While the worst separation must be lower jointly over two databases than over each one separately, Table 3.5 shows that this downgrade of separation is in our case only small and the mouth is found very reliably. The new database turns out to be more problematic than the original one, it contains one problematic mouth, which at the same time influences the result of all the optimization procedures over the whole new database.

3.7 Preliminary Transformations of the Data

The spatial autocorrelation in images of faces is clearly large. The Moran's coefficient I defined by (2.6) computed for the mouth in the left image in Figure 2.8 equals 0.90 or computed for the bearded mouth template (Figure 2.2) is equal to again 0.90. In this chapter we try to transform both the template and the image to remove the spatial autocorrelation.

We consider the following transformation. Let $\mathbf{A} = (a_{ij})$ be a matrix with m rows and n columns. We define the matrix $\mathbf{B} = (b_{ij})$ with $m-1$ rows and $n-1$ columns by

$$b_{ij} = 2a_{i+1,j+1} - a_{i,j+1} - a_{i+1,j}, \quad i = 1, \ldots, m-1, \quad j = 1, \ldots, n-1. \tag{3.11}$$

The bearded template after this transformation is shown in Figure 3.19. We can see that the top lip obtains low grey values, the bottom lip large values and the rest of the rectangle is more homogeneous than the original template. After the transform the whole top lip rather than just its boundary has large values. The reason is that the original top lip is the darkest on the horizontal segment in its bottom part. The horizontal segments above it are gradually lighter as the distance from the bottom increases. Therefore the transform takes into account these steps between rows and the top lip becomes almost homogenous but with a different value than

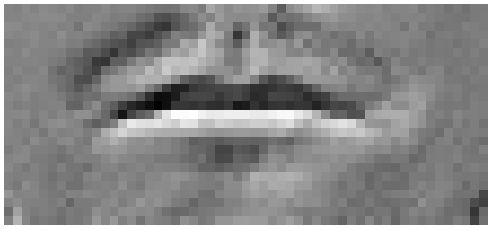

Figure 3.19: The bearded mouth template after the transformation (3.11).

Figure 3.20: Figure 1.1 after the transformation (3.11).

the neighbourhood of the mouth. The Moran's coefficient I of Figure 3.19 equals 0.70, so the two-dimensional autocorrelation is still large but lower than in the original bearded template.

Figure 1.1 after the transform is shown in Figure 3.20. The face has now about the same grey values as the background which may be a disadvantage in locating landmarks. In Figure 3.20 the boundaries of the man and his landmarks are outlined. The transform (3.11) emphasizes the boundaries although it is not directly a boudary extraction technique and we can expect locating the mouth with templates in the transformed image to be difficult. A general criticism of using templates for boundaries is given for example by Grenander (1993) and Winkler (1995). These books criticize that template matching does not take into account variability of individual objects from ideal boundaries. Templates work reliably for objects with the same shape, size and rotation, but boundaries report a too large variability. Although the images after the transform (3.11) may seem to be too sensitive to deformations and local deviations in order to use templates, they allow the mouth to be located surprisingly well as we will now see.

The transformation (3.11) subtracting two neighbours from each pixel is carried out on every of the 124 images from the training set. The template from Figure 3.19 is used to locate the mouth. Using equal weights, the mouth is located correctly in all 124 images and the worst separation equals 1.04.

Applying the analytical search with $c = 0.005$ and equal initial weights increases the worst separation to 1.70. The solution is shown in the left part of Figure 3.21. No pixel has a zero weight and the maximal weight equals about 0.0017. The largest weights can be found in grey areas above the lips below the moustache or in the bottom part of the rectangle. Sorted values

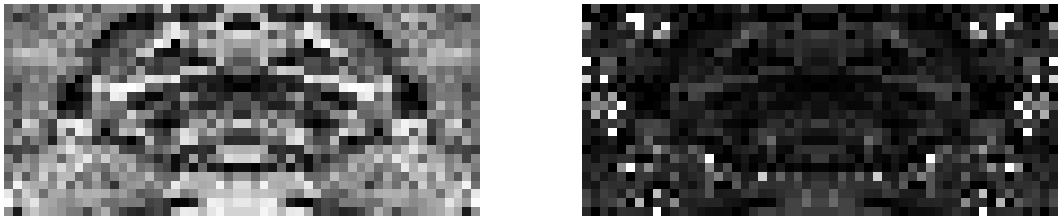

Figure 3.21: Optimal weights for images transformed by (3.11). Starting with equal weights, the analytical (left) and then the approximative (right) search has been applied.

Table 3.6: Results of locating the mouth using the transform (3.11) and starting with equal initial weights. The weights are optimized over one database and then applied to the other of the two databases.

Approach	Optimize over original database	Apply on new database	Optimize over new database	Apply on original database
Equal	1.04	0.94	1.04	0.94
Analytical	1.70	1.20	1.65	1.04
Anal. $\Longrightarrow$ approx.	1.89	1.28	1.73	1.06

of the weights are almost uniformly spread over the interval between the minimal and maximal one.

The approximative search with the constraint $c = 0.005$ was then applied on the solution of the analytical search. The worst separation further increases to 1.89. This is only slightly less than the best result obtained without the transformation (3.11), which was 1.94. The solution is shown in the right part of Figure 3.21. The upper bound is attained in 24 pixels which correspond mainly to the sides of the template. In other 7 % of pixels the weight is zero. These are scattered more or less in a homogenous way over the template.

We remark that radial weights do not work as well as equal weights after the transformation (3.11). The mouth is found correctly in 98 % of the images and the worst separation 0.85. We believe that the transform (3.11) reduces a lot of information and a further downweighting of pixels farther away from the midpoint would be excessive, so we consider the equal weights to be more appropriate as initial ones.

Further we have optimized over the new database of 88 images after the transformation (3.11) performed on the images and also on the bearded template. Equal weights locate the mouth in 99 % of images and the worst separation equals 0.94.

The result of the analytical search is very similar to the left image in Figure 3.21, which was optimized over the original database. The worst separation is improved to 1.65. The weights locate the mouth correctly in 98 % of images and the worst separation equals 0.85. The maximal weight equals 0.0022. There is a zero weight in about 5 % of pixels.

The approximative search with $c = 0.005$ applied on the solution of the analytical search improves further the worst separation to 1.73. There is a remarkable similarity of the solution with the right part of Figure 3.21. The maximal weight is attained in 22 pixels and a zero weight is present in 17 % of the pixels. Optimization over the new database does not improve

the worst separation as much as over the original database. Although the new database contains less images, one of its mouths is more problematic as mentioned at the beginning of Chapter 3.6.

The results above have been optimized either over the original database with 124 images or over the new database with 88 images. We now apply the resulting weights always to the other database to check the reliability of the method. The results are shown in Table 3.6. The first row shows the equal weights themselves, which are used as initial ones. We repeat that the solution of the analytical search optimized over the original database gives the worst separation 1.70. Applying this solution on the new database, the separation equals 1.20.

Optimizing over the original database using the analytical and then approximative search, the worst separation in the original database equals 1.89. This is shown in the bottom row of Table 3.6. These weights applied to the new database give the worst separation 1.28. This result is better than any method without the transformation (3.11). The best result in the new database (without optimizing over it) in Table 3.5 was 1.16. Although our attempt to remove the spatial autocorrelation could reject too much information from the data, it seems actually to bring benefits.

Results optimized over the new database and applied to the original database are however not so good and 1.06 is here the worst separation when the analytical and approximative search are both used. The new database is perhaps too small to serve as a training set.

The transformation (3.11) resembles a standard technique to remove autocorrelation from one-dimensional time series. Let us denote such a time series by $x_1, x_2, \ldots, x_n$. These data are often transformed to $x_t - \hat{\rho} x_{t-1}$, where $\hat{\rho}$ stands for an estimator of the autocorrelation coefficient. This is the idea for example of the Cochrane-Orcutt transformation (Greene 1993) which is then followed by computing the least squares regression for the transformed data.

This has motivated us to replace (3.11) by

$$b_{ij} = 2a_{i+1,j+1} - ka_{i,j+1} - ka_{i+1,j}$$

and compute the worst separation with different choices of k. The worst separation is the largest for $k = 0.85$ and namely equals 1.17. We recall that the worst separation with $k = 1$ equals only 1.04.

Other possible transformations do not work so well. Instead of comparing each pixel with its left and top neighbours in (3.11), it is possible to compare each pixel with its four neighbours (top, bottom, left, right) or in an analogous way eight neighbours (the same and diagonal ones). Then the worst separation attains 0.76 for four neighbours and 0.72 for eight neighbours. Although such transformations use more information about the neighbourhood of each pixel, they do not actually remove the spatial autocorrelation and therefore there is no reason for such transforms to bring improvements.

3.8 Robustness of the Results

Robustness to a Plaster

Our aim is to examine the local sensitivity of different weights. We modify the mouth in a small number of pixels and use the bearded template (Figure 2.2) together with different weights to search for the mouth. An illustration is shown in Figure 3.22 where grey values of 15 pixels are set to 1. First we examine the effect of the plaster in one example and then we put the plaster

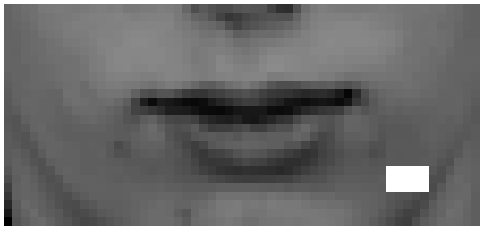

Figure 3.22: A mouth with a plaster.

Table 3.7: The effect of a plaster in the mouth and nonmouth from Figure 2.5.

Weights	$r_W(\boldsymbol{x},\boldsymbol{t})$	$r_W(\boldsymbol{x}^*,\boldsymbol{t})$	$r_W(\boldsymbol{z},\boldsymbol{t})$	$\frac{r_W^F(\boldsymbol{x},\boldsymbol{t})}{r_W^F(\boldsymbol{z},\boldsymbol{t})}$	$\frac{r_W^F(\boldsymbol{x}^*,\boldsymbol{t})}{r_W^F(\boldsymbol{z},\boldsymbol{t})}$
Equal	0.48	0.40	0.52	0.90	0.74
Radial	0.66	0.63	0.38	1.97	1.85
Analytical	0.55	0.52	0.28	2.18	2.01
Approx. unconstrained	0.63	0.24	0.26	2.79	0.92
Approx. $c = 0.02$	0.70	0.32	0.31	2.71	1.67
Approx. $c = 0.00$	0.65	0.31	0.34	2.19	1.55
Approx. $c = 0.005$	0.68	0.48	0.35	2.27	2.17
Anal. $\Longrightarrow$ approx.	0.56	0.49	0.30	2.04	1.71

to each mouth in the whole database and again compare the performance of different weights in locating the mouth.

We examine the effect of a plaster in the mouth shown in Figure 2.5. We place the plaster of the size 3×5 pixels each time on a different position. For the solution of the unconstrained approximative search (Figure 3.4) the position of the plaster is shown in Figure 3.22, which covers the most highly influential pixels. For other choices of weights we place the plaster also to the bottom right corner, but we have selected such positions where it covers as many pixels with the largest weights as possible. The grey values in the plaster were set to 1. We have computed the weighted correlation between this mouth with a plaster and the bearded template with different weights.

Table 3.7 compares robustness of different weights to the plaster. There we use the notation $\boldsymbol{t}$ for the bearded template, $\boldsymbol{z}$ for the nonmouth from Figure 2.5, $\boldsymbol{x}$ for the mouth from the same image and finally $\boldsymbol{x}^*$ for the same mouth with the plaster (Figure 3.22). The table compares the weighted correlation between the mouth and the template, between the mouth with the plaster and the template and between the nonmouth and the template. The last two columns contain the separation between $\boldsymbol{x}$ and $\boldsymbol{z}$ and between $\boldsymbol{x}^*$ and $\boldsymbol{z}$. The weights include equal and radial ones, the solutions of the unconstrained and constrained approximative searches with different values of the upper bound and analytical and two-stage searches.

We can observe the plaster to decrease the weighted correlation of the mouth with the template rapidly. The solution of the unconstrained approximative search is influenced by the plaster remarkably strongly. The constrained optimization with $c = 0.005$ seems to be more robust. In this example the radial weights are not influenced so much by the plaster which is

situated quite far from the midpoint and thus get lower weights. We have found other examples (another mouth and nonmouth), where the radial weights do not find the mouth correctly when the plaster is included.

To summarize this example, radial weights report a too large weighted correlation between the nonmouth and the template, so there is also a too low separation of the mouth from nonmouths. Moreover weights with highly influential pixels are too sensitive to small changes of the data. The solutions of the constrained searches turn out to be reasonably robust and can be recommended for practical using.

Without any surprise the degenerated solution of the unconstrained approximative search is strongly influenced by the plaster. However when the template is placed a few pixels aside, only low weights correspond to the plaster and the mouth can be expected to be located correctly. Although the lips are then shifted aside compared to their position in the template, the result turns out to be satisfactory. This will be now documented.

We place a small plaster in the neighbourhood of the mouth in every of the 124 images in the database in a similar way as before. The plaster is placed always on the same position compared to the position of the midpoint of the mouth. To be specific, the plaster is situated below the midpoint of the mouth by 7 to 9 rows and on the right from the midpoint by 16 to 20 columns.

Here the position of the plaster corresponds to highly influential pixels of the unconstrained approximative search. For other weights the plaster covers also some of the pixels with the weight equal to the upper bound, but maybe the position of the plaster is not the worst possible.

We have searched for the mouth using all choices of weights mentioned in Table 3.7. Equal weights fail in locating the mouth in some pictures, namely locate the mouth correctly in 88 % of pictures. Each of the remaining choices of weights is robust to such plaster and locates the mouth correctly in 100 % of pictures. This robustness surprisingly holds also for the solution of the unconstrained approximative search. These results can be explained by the robustness of the template matching to a small shift.

Also the following study is devoted to a small shift of images. We consider two images shifted from each other by one pixel. We take the left image from Figure 2.8, which is the mouth of dr. Böhringer (Figure 1.1). The weighted correlation is computed between this mouth and the mouth taken again from Figure 1.1 but now shifted by one pixel. Using all choices of weights listed in Table 3.7, the weighted correlation between the mouth and itself shifted aside is between 0.91 and 0.96 for different weights. By shifting upwards or downwards the weighted correlation is between 0.81 and 0.89. This robustness follows from the large two-dimensional autocorrelation of the images.

The results of the constrained searches turn out to be robust to the plaster, in other words to big changes of a small number of grey values. Moreover they are robust to a small shift of the template with respect to the mouth.

Robustness to Nonsymmetry

The whole Chapter 3 considers weights symmetric along the vertical axis through the middle. Although every mouth in our database turns out to be reasonably symmetric, it is desirable that the weights are not sensitive to a possible nonsymmetry of the mouth. Only such weights would then work reliably in locating the mouth in different images.

In order to study the robustness of the methods to nonsymmetry we have changed the grey values in the mouth areas to make them nonsymmetric. An example is shown in Figure 3.23.

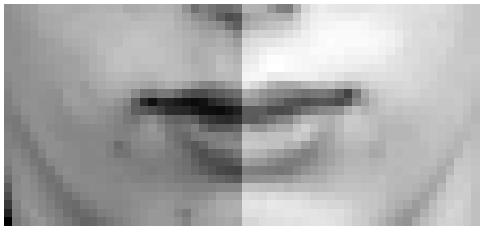

Figure 3.23: The mouth from Figure 2.5 modified by $\varepsilon = 0.1$ is used to examine the robustness of the methods to nonsymmetry.

Table 3.8: Effect of the nonsymmetry in the example from Figure 2.5. The separation between the mouth and nonmouth, where grey values of the mouth are increased in its right half by different values of ε.

Weights	$\varepsilon = 0$	$\varepsilon = 0.05$	$\varepsilon = 0.10$	$\varepsilon = 0.15$
Equal	1.06	1.04	0.88	0.73
Radial	1.23	0.88	1.19	1.05
Analytical	2.18	2.01	1.67	1.32
Approx. unconstrained	3.50	2.90	1.86	1.25
Approx. $c = 0.02$	3.04	2.70	2.01	1.47
Approx. $c = 0.01$	2.20	1.88	1.41	1.04
Approx. $c = 0.005$	1.66	1.58	1.34	1.12
Anal. $\Longrightarrow$ approx.	2.04	1.88	1.55	1.22

First we describe the experiments with the mouth and nonmouth from Figure 2.5. Then we modify the mouth in every image to become nonsymmetric and compare the performance of different weights in locating the mouth.

We consider the mouth and nonmouth from Figure 2.5. In previous parts of Chapter 3 we have used the bearded template to discriminate between mouths and nonmouths. To study the effect of nonsymmetry on the described methods we increase grey values in the right half of the mouth in Figure 2.5 by a constant, say ε. The severity of the modification of a mouth by $\varepsilon = 0.10$ is shown in Figure 3.23. However there we see the left part darker than in Figure 2.5, because the graphical output scales images to the interval $[0, 1]$.

The results of this example are given in Table 3.8. The first column with $\varepsilon = 0$ corresponds to no modification of the mouth. Different rows correspond to different weights, which include the solutions of the analytical search, then the unconstrained and constrained approximative searches with different values of the upper bound on the weights and finally the two-stage search. All of these weights have been optimized over the original database with 124 images starting with radial initial weights. The values in the first column are the values of the separation between the mouth and nonmouth shown in Figure 2.5. The next three columns use the modification of the mouth, namely the grey values in the right half of the mouth have been increased by 0.05, 0.10 and 0.15 respectively. The nonmouth was not modified.

There is no monotone relationship between the value of ε and the value of the separation. Equal and radial weights fail while the results of the analytical and/or approximative searches

Figure 3.24: Study of robustness to nonsymmetry of the mouth. Grey values in a half of the mouth area increased by 0.10 (left) and 0.15 (right).

turn out to be robust to nonsymmetry. Therefore the requirement on the optimal weights to be symmetric turns out not to be restrictive.

In a similar manner we modify every mouth in the database of 124 images and use the bearded template and different choices of weights to locate the mouth. An example of the modified picture is shown in Figure 3.24, namely with $\varepsilon = 0.10$ on the left and 0.15 on the right.

The results are summarized in Table 3.9. For each of the weights the first row gives the worst separation with the particular weights and the next row gives in parentheses the percentage of images, in which the mouth is located correctly. For example the radial weights locate the mouth correctly in every image with $\varepsilon = 0$ and the worst separation equals 1.11, which decreases to 0.84 for $\varepsilon = 0.15$ and then the mouth is located correctly only in 91 % of images.

The modification of the mouth with $\varepsilon = 0.15$ is already quite a severe alteration of the original mouth. Table 3.9 documents that the solution of the unconstrained approximative search is very sensitive. Better robustness properties are obtained with constrained approaches and the best results are obtained with the two-stage search. These weights (left part of Figure 3.12) locates the mouth in all 124 images correctly and the separation between the worst mouth and nonmouth in that case equals 1.07.

We now examine theoretically the effect of the modification of one half of the mouth. We assume the mouth to be symmetric and denote its grey values after being transformed to a vector by $\boldsymbol{x} = (x_1, \ldots, x_n)^T$. Let the weights $w_1, \ldots, w_n$ fulfill $\sum_{i=1}^n w_i = 1$. We assume that the mouth contains an even number of columns. One half of the grey values of the modified mouth $\boldsymbol{x}^*$ are equal to those from $\boldsymbol{x}$ and the rest are the same values increased by ε.

The weighted mean in the original mouth $\bar{x}_W$ and the (sample) weighted variance $S_W^2(\boldsymbol{x}; \boldsymbol{w})$ are defined by

$$\bar{x}_W = \sum_{i=1}^n w_i x_i \quad \text{and} \quad S_W^2(\boldsymbol{x}; \boldsymbol{w}) = \sum_{i=1}^n w_i (x_i - \bar{x}_W)^2.$$

In the modified mouth the weighted mean and weighted variance can be easily shown to be

Table 3.9: Effect of the nonsymmetry in the whole database of 124 images. Results of locating the mouth, where grey values of every mouth are increased in its right half by different values of ε.

	Top: the worst separation over the whole database. (Below: percentage of images with correctly located mouth.)			
	ε=0	ε=0.05	ε=0.10	ε=0.15
Equal	0.78 (0.94)	0.71 (0.90)	0.61 (0.69)	0.49 (0.33)
Radial	1.11 (1.00)	1.04 (1.00)	0.90 (0.98)	0.84 (0.91)
Analytical	1.68 (1.00)	1.52 (1.00)	1.28 (1.00)	1.02 (1.00)
Approx. unconstrained	2.10 (1.00)	1.50 (1.00)	0.99 (0.99)	0.75 (0.82)
Approx. $c = 0.005$	1.44 (1.00)	1.33 (1.00)	1.12 (1.00)	0.94 (0.98)
Anal. $\Longrightarrow$ approx.	1.73 (1.00)	1.55 (1.00)	1.29 (1.00)	1.07 (1.00)

equal to

$$\bar{x}^*_W = \bar{x}_W + \frac{\varepsilon}{2} \quad \text{and} \quad S^2_W(\boldsymbol{x}^*;\, \boldsymbol{w}) = S^2_W(\boldsymbol{x};\, \boldsymbol{w}) + \frac{\varepsilon^2}{4}.$$

Clearly both values increase compared to the mean of the grey values and weighted variance of the original mouth.

Let $\bar{t}_W$ denote the weighted mean of the template $\boldsymbol{t}$. Let $S_W(\boldsymbol{x},\, \boldsymbol{t};\, \boldsymbol{w})$ denote the (sample) weighted covariance between the mouth and the template. This is defined by

$$S_W(\boldsymbol{x},\, \boldsymbol{t};\, \boldsymbol{w}) = \sum_{i=1}^{n} w_i(t_i - \bar{t}_W)(x_i - \bar{x}_W)$$

Assuming the symmetry of the mouth, template and the weights, the weighted covariance between the template and the modified mouth equals the weighted covariance between the template and the original mouth $S_W(\boldsymbol{x},\, \boldsymbol{t};\, \boldsymbol{w})$. The conclusion is the formula for the (sample) weighted correlation between the template and the modified mouth

$$r_W(\boldsymbol{x}^*,\, \boldsymbol{t};\, \boldsymbol{w}) = \frac{S_W(\boldsymbol{x}^*,\, \boldsymbol{t};\, \boldsymbol{w})}{S_W(\boldsymbol{x}^*;\, \boldsymbol{w}) S_W(\boldsymbol{t};\, \boldsymbol{w})} = r_W(\boldsymbol{x},\, \boldsymbol{t};\, \boldsymbol{w}) \frac{S_W(\boldsymbol{x};\, \boldsymbol{w})}{S_W(\boldsymbol{x}^*;\, \boldsymbol{w})} =$$

$$= r_W(\boldsymbol{x},\, \boldsymbol{t};\, \boldsymbol{w}) \frac{S_W(\boldsymbol{x};\, \boldsymbol{w})}{\sqrt{S^2_W(\boldsymbol{x};\, \boldsymbol{w}) + \frac{\varepsilon^2}{4}}}.$$

Now we come back to the example with the mouth and nonmouth of Figure 2.5. The modification of the mouth has a strong effect with a large ε and mainly in mouths with a small value of the weighted variance. The mouth has the weighted standard deviation (square root of the weighted variance) for all possible choices of weights from Table 3.8 between 0.008 and 0.009.

Only with equal weights the value is 0.012. The weighted correlation between the template and the modified mouth decreases compared to that between the template and original mouth. Using the theoretical result, this level of downtrend should be very similar for all choices of weights (except for the equal ones). Therefore the weights with a large separation between the mouth and nonmouth keep a large separation even in the modified nonsymmetric case.

However the mouth in Figure 2.5 is not well symmetric, which explains some differences between the theoretical result and the values in Table 3.8. Mainly the radial weights differ from the theoretical results and the reason is a local nonsymmetry in the midpoint of the mouth, which has large weights. We remark that this chapter does not examine the sensitivity of the methods to nonsymmetry in a small amount of pixels, which however have highly influential weights.

Robustness to Size and Rotation

Now we examine the robustness of different choices of weights from Chapter 3 to a different size and rotation of faces. Just like in Chapter 3.5 we transform the 124 images to the size 173×230 pixels. In other words the height and width of images is reduced by 10 %, while the number of the pixels correspond only to 81 % of the original number of pixels. Below we describe these smaller images as smaller by 10 %, because the reduction of segments is important for the template matching and not the areas.

The bearded template in its original size 26×56 pixels is applied to search for the mouth. The results for different weights are given in Table 3.10. For each choice of weights the first row gives the worst separation between the mouth and worst nonmouth over the whole database of images. Below in parentheses we give the percentage of images in which the mouth is located correctly. The table has three vertical parts. Original images are considered in the left one. The middle part contains results for images with the smaller size by 10 %. The result of the analytical search and the result of the two-stage search (first analytical and then approximative) are the only robust weights, for which the mouth is located correctly in all images. The two-stage search has the worst separation 1.03 over the database of images with the smaller size.

The right part of Table 3.10 shows results of locating the mouth in images rotated by $+10^\circ$ and also -10°. On the whole we can say that equal weights do not perform well at all and radial ones are too sensitive to both size and rotation. The result of the two-stage search is robust to size and rotation. Although the worst separation exceeds 1 only slightly in both cases, we must keep in mind that the mouth rotated by $\pm 10^\circ$ is very difficult to be located with a nonrotated template. To summarize, the weights obtained by the two-stage search of Chapter 3.5 turn out to be robust not only to local changes of grey values but also to nonsymmetry. This together with their good performance in locating the mouth are reasons why they can be recommended for practical usage.

3.9 Optimizing the Weights for the Eyes

In this chapter we search for the eyes using the template from the very left image of Figure 2.3 of the size 26×29 pixels. This is a template for the right eye of each person, which is therefore situated on the left side of the image. The weights for the weighted correlation coefficient between the template and parts of the image are optimized in this chapter. Again the two-stage search is used and results with different initial weights are presented.

Table 3.10: Robustness of locating the mouth to different size or rotation of the face. Percentages of correctly located mouths with different choices of weights for the template.

	Top: the worst separation over the whole database. (Below: percentage of images with correctly located mouth.)					
	Standard images		Smaller by 10 %		Rotated by $\pm 10^\circ$	
Equal	0.78		0.46		0.60	
		(0.94)		(0.78)		(0.64)
Radial	1.11		0.85		0.93	
		(1.00)		(0.99)		(0.98)
Analytical	1.68		1.13		0.99	
		(1.00)		(1.00)		(0.99)
Approx. unconstrained	2.10		0.83		1.06	
		(1.00)		(0.99)		(1.00)
Approx. $c = 0.005$	1.44		0.82		1.03	
		(1.00)		(0.98)		(1.00)
Anal. $\Longrightarrow$ approx.	1.73		1.03		1.03	
		(1.00)		(1.00)		(1.00)

Table 3.11: Worst separation obtained with different weights of Figures 3.25 and 3.26 in locating the eyes with the templates shown in the same figures.

	Search for the right eye					
	Initial		Analytical		Two-stage	
Weights	R	L	R	L	R	L
Radial	0.40	0.94	1.23	0.95	1.27	0.98
	(0.93)	(0.98)	(1.00)	(0.99)	(1.00)	(0.99)
Equal	0.43	0.78	0.86	0.69	1.03	0.82
	(0.93)	(0.93)	(0.87)	(0.90)	(1.00)	(0.94)
Random	0.41	0.78	0.91	0.78	1.10	0.92
	(0.94)	(0.93)	(0.89)	(0.93)	(1.00)	(0.97)
Random	0.45	0.75	0.87	0.69	1.02	0.78
	(0.93)	(0.92)	(0.89)	(0.88)	(1.00)	(0.92)
	Search for the left eye					
	Initial		Analytical		Two-stage	
Weights	L	R	L	R	L	R
Radial	0.94	0.40	1.23	0.61	1.28	0.64
	(0.98)	(0.93)	(1.00)	(0.99)	(1.00)	(0.99)
Equal	0.78	0.43	1.34	0.45	1.36	0.46
	(0.93)	(0.93)	(1.00)	(0.98)	(1.00)	(0.99)
Random	0.79	0.45	1.29	0.49	1.36	0.54
	(0.93)	(0.94)	(1.00)	(0.98)	(1.00)	(0.99)
Random	0.78	0.43	1.34	0.54	1.34	0.54
	(0.93)	(0.93)	(1.00)	(0.99)	(1.00)	(0.99)

To optimize the weights for the eye template we use the approach of Chapter 3.5, which is a general method for any template. The only difference is that now we do not require the weights to be symmetric. Independently on the search for the right eye we will search for the left eye using the mirror reflection of the same eye template. Finally the results will be cross-validated in the following way. Both the template for the right eye and the optimal weights are mirror reflected and applied to search for the left eye. Similarly the template and weights for the left eye are used to search for the right eye, again after a mirror reflection.

The template for the right eye is compared with every rectangular part of the image with size of the same size 26×29 pixels using the weighted correlation with different choices of weights. We consider the result of the search to be successful if the midpoint of the most suspicious area does not have its distance from the midpoint of the right eye larger than five pixels. In other words the left eye is considered to be also a noneye in this context.

Results with different initial weights are presented in Figure 3.25 and Table 3.11. The template is shown at the top of the figure. Its next four rows correspond to the initial weights (left), then the result of the analytical search (middle) and finally the two-stage search (right), which is obtained by applying the constrained approximative search on the solution of the analytical search. For all cases there was the upper bound on the weights $c = 0.005$ required.

The first two rows of Table 3.11 are devoted to radial weights. In the very left "R" column there is the worst separation 0.40 and below in parentheses the success rate 0.93, which means that the eye is located in 93 % out of the 124 images of the whole database. The worst separation is a very low number obtained in one problematic right eye, which contains dark hair in its nearest neighbourhood. This hair belong to the area, which is compared with the eye template. The next column entitled "L" shows the result of the cross-validation (locating the left eye), which will be explained later.

The analytical search is applied using Chapter 3.2 without the condition on the symmetry of the weights. Its results depend on the initial weights. Starting with radial weights the analytical search improves the worst separation to 1.23 and mainly the pixels in the central area obtain large weights.

Then the approximative search is applied using Chapter 3.4 again without the condition on the symmetry of the weights. The condition (3.9) has now the form $w^* \leq c(1+\delta) - \delta$. One pixel is selected at the time from the whole template of the size 26×29 pixels and in the new weights the weight is modified only in this pixel while retaining the sum of the weights to be 1. The worst separation further increases to 1.27, which is shown in the top row of Table 3.11 in the corresponding "R" column. The largest weights of this solution are present again in the middle area of the template. There is however a larger number of pixels with the largest weight 0.005 in the solution of the two-stage search, namely 9 % of pixels compared to 3 % of pixels in the solution of the analytical search.

The template finds the right eye correctly in 93 % out of the 124 images with equal weights and the worst separation equals 0.43. The solution of the analytical search with equal initial weights locates the right eye only in 87 % of images and finally the approximative search improves the worst separation above 1, namely to 1.03.

Altogether there are radial, equal and two choices of random initial weights used in Figure 3.25. The random weights are generated from a uniform distribution over the interval $[0; 0.0028]$ so that their sum equals 1. The solutions starting with random weights are very similar to each other, larger weights are situated in the neighbourhood of the eye and the worst separation obtained with the solution of the two-stage search attains 1.10 and 1.02 for different

initial weights. The best result is obtained for radial initial weights.

Now we search for the left eye using the mirror reflection of the previous right eye template. This search is independent from the previous search for the right eye. The template is shown at the top of Figure 3.26 and the results with different initial weights are presented in the same figure. The initial weights are the same as in Figure 3.25, namely radial, equal and two choices of random ones. The worst separation over the whole database lies between 0.78 and 0.94.

The worst separation obtained with the initial weights, solution of the analytical and two-stage searches are summarized in the bottom half of Table 3.11 and the results are always in the "L" columns, namely the worst separation and below in parentheses the percentage of images in which the left eye is located correctly.

The template for the right eye after the mirror reflection locates the left eye with radial weights in 98 % of cases and the worst separation equals 0.94. The weights obtained by the analytical search are shown in Figure 3.26 in the second row in the middle and the worst separation equals 1.23. The approximative search further improves the worst separation to 1.28 and the weights are shown in the right column of the second row of the same figure.

The template together with equal weights locate the left eye in 93 % of images of the given database. The analytical search improves the worst separation far beyond 1 and then the approximative search does not modify the solution of the analytical search much, because it is probably already quite close to the global optimum.

Also starting with random weights the analytical search improves the worst substantially and the approximative search changes the solution of the analytical search only slightly. The solutions obtained with the two different choices of random weights are remarkably similar to each other.

Comparing the solution obtained for the right and left eye template, the resulting weights look in a different way and the worst separation is much larger for the left eye. Also all initial weights perform better for the left eye. One of the right eyes in the database is namely problematic because of hair in its the nearest neighbourhood of the right eye. The left eye of that person is however quite standard and there is no problematic left eye in the whole database of images. The two-stage search starting with any initial weights locates the left eye correctly with the worst separation between 1.28 and 1.36, which is better than for the right eye.

Finally we have applied the results optimized over one eye to locating the other eye, of course after mirror reflecting the template and the weights. These results are shown also in Table 3.11. We have already described the results of searching for the right eye and optimizing the weights for its template. These are given in the top half of Table 3.11 in the "R" columns. Now these weights are mirror reflected and are applied together with the mirror reflection of the template to search for the left eye. These results are presented there in the "L" columns.

For example the two-stage search starting with radial weights gives the worst separation 1.27 in the search for the right eye and only 0.98 when the right template and these weights are used after a mirror reflection to search for the left eye. In that case the left eye is found in 99 % of cases. This is still better than the performance rate 98 % obtained with radial weights in the search for the left eye. The difference in locating the left eye is however only slightly better than with initial weights.

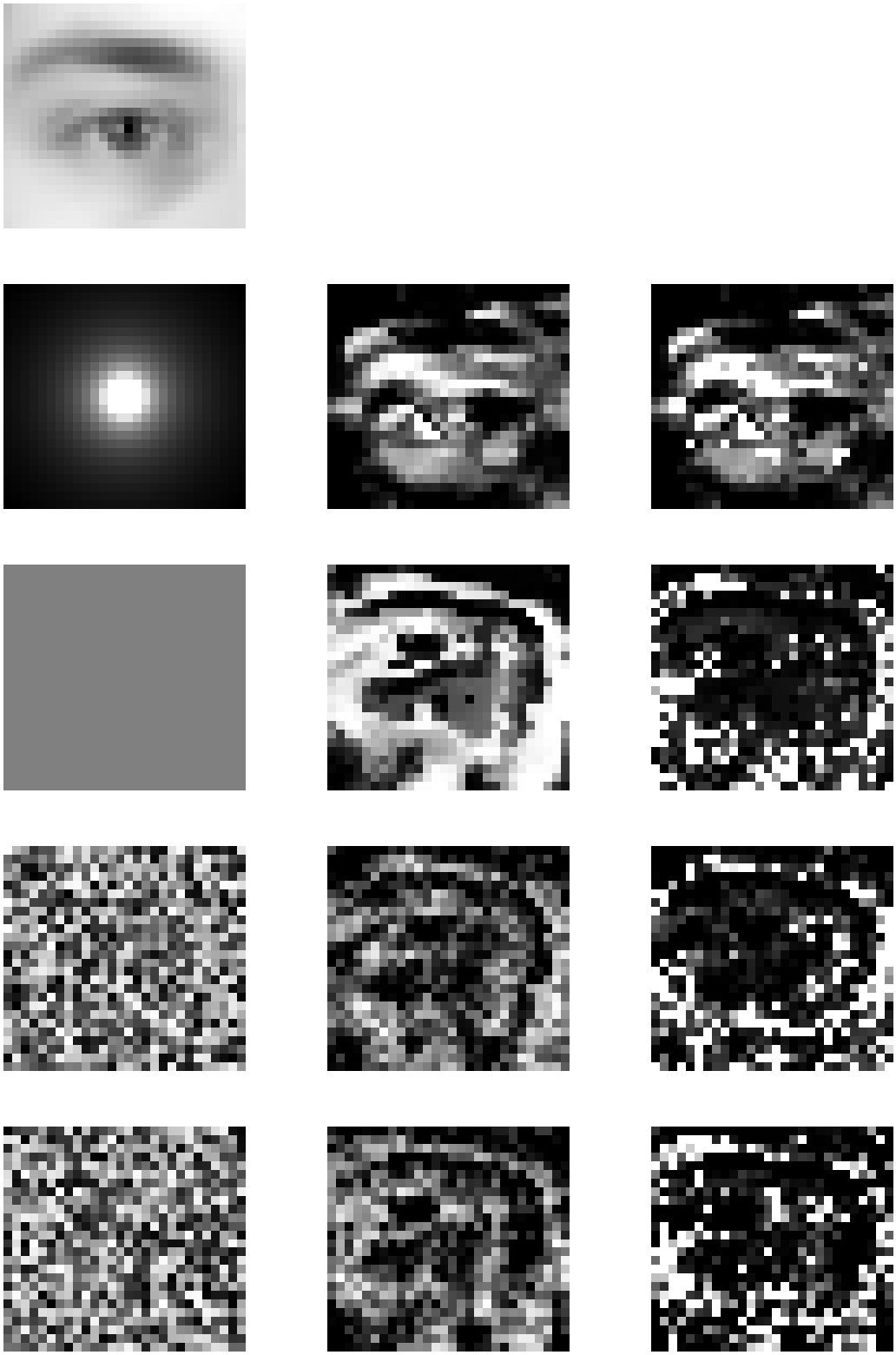

Figure 3.25: Above: a template for the right eye. Below: different initial weights (left), result of the analytical (middle) and two-stage search (right).

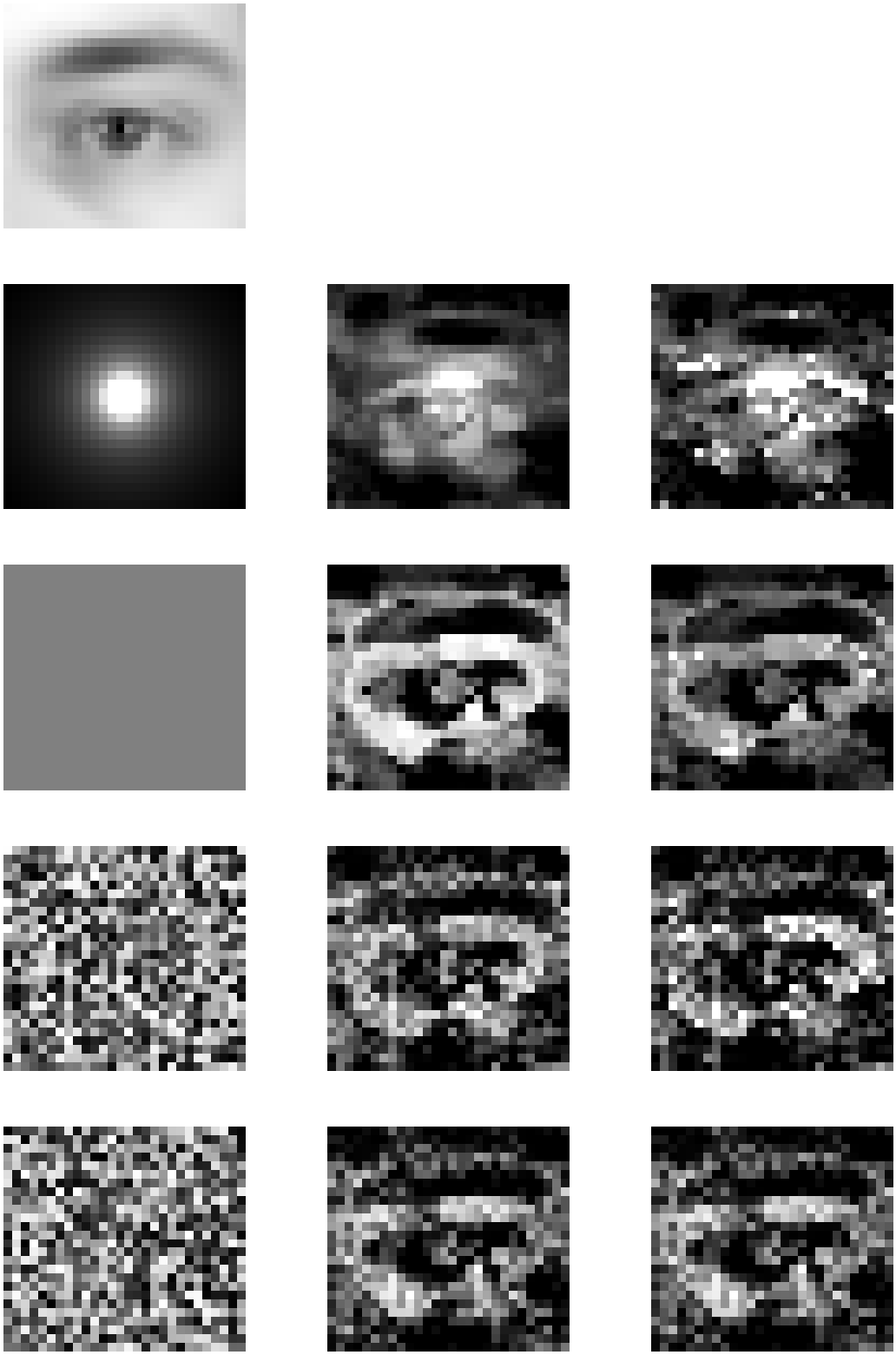

Figure 3.26: Above: a template for the left eye. Below: different initial weights (left), result of the analytical (middle) and two-stage search (right).

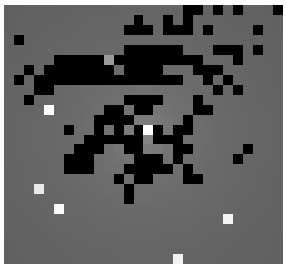

Figure 3.27: Weights obtained as a result of the approximative search with $c = 0.30$ for the template from Figure 3.25 and radial initial weights.

The optimized weights for the right eye template are rather weak because the optimization is strongly influenced by the problematic worst right eye. The results are weak when applied to the search for the left eye and we can observe that one atypical eye plays a negative role in locating typical eyes.

In the bottom half of Table 3.11 there is the opposite situation, namely the weights for the left eye template are optimized and after the mirror reflection applied to search for the right eye. The results are given in the "R" columns. Using the weights obtained by the two-stage search, the right eye is located in 99 % of images independently on the initial weights. This can be considered very successful, because the right eye is located in all images except for the one exceptional with the hair in its neighbourhood. There is no problematic left eye in the database, the optimal weights for the left eye contain typical features of an eye and after the mirror reflection perform well for all typical right eyes as well.

Furthermore we present the result of the unconstrained approximative search. In Chapter 3.3 we have seen that its solution tends to degenerate for the mouth template and now the situation for eye templates is analogous. We use the template for the right eye (Figure 3.25 at the top) and radial weights. Instead of the unconstrained approximative search we apply the constrained version with a very large upper bound $c = 0.3$ to prevent the computation from collapsing. As the algorithm iterates, the weights in some pixels increase dramatically. The resulting weights are shown in the log scale in Figure 3.27. The upper bound 0.30 is attained in two pixels, one of them is the white pixel almost precisely in the midpoint of the rectangle and the other is situated near the left side of the rectangle. The weight is exactly zero in the black pixels. The worst separation over the whole database equals only 0.73. This is attained in one image of a man and the most suspicious area is the top of the ear containing dark hair in its middle, the face on one side and a lighter background on the other side.

This confirms the difficulty with optimizing a complicated function. The optimum corresponding to a large value of the upper bound is theoretically better than optimum under stricter constraints, but the algorithm increases the weights in some pixels very quickly too far to the upper bound and then does not come out from the obtained local optimum. Better results are therefore obtained with the much slower approach binding the weights from above by a relatively small upper bound.

Table 3.12: Worst separation over the database of images obtained with different optimal weights and optimal templates.

Init.	Init.	Init.	Weights		Template	
weights	template	worst separ.	Anal.	Approx.	Anal.	Approx.
Radial	Bearded	1.11	1.68	1.73	1.75	1.75
Equal	Bearded	0.78	1.67	1.94	1.94	1.94
3.	Bearded	0.91	1.33	1.88	2.07	2.07
4.	Bearded	0.80	1.52	1.97	2.00	2.00
5.	Bearded	0.82	1.54	1.85	2.16	2.16
Init.	Init.	Init.	Template		Weights	
weights	template	worst separ.	Anal.	Approx.	Anal.	Approx.
Radial	Bearded	1.11	1.79	1.79	1.83	1.90
Equal	Bearded	0.78	2.12	2.12	2.21	2.29
3.	Bearded	0.91	2.05	2.05	2.06	2.06
4.	Bearded	0.80	2.06	2.06	2.21	2.24
5.	Bearded	0.82	2.19	2.19	2.20	2.30

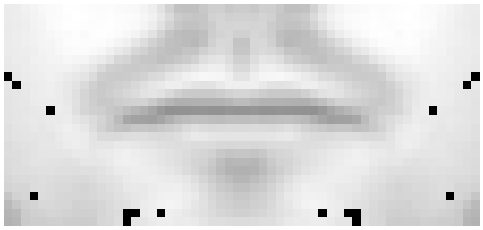

Figure 3.28: Optimal template obtained with optimal weights (top right corner of Figure 3.12) and bearded initial template.

3.10 Optimization of the Template Itself

So far all approaches in Chapter 3 retain the template and optimize the weights for the weighted correlation coefficient. The separation (3.1) can be also viewed as a function of the template with fixed weights. The Taylor's expansion for the separation $f(\boldsymbol{w}, \boldsymbol{x}, \boldsymbol{z};\ t_1+\delta_1, \ldots, t_n+\delta_n)$ can be considered in an analogous way with (3.4). Then it is possible to apply both the analytical and approximative search to approximate to the optimal template. All of the approaches below are constrained, namely in optimizing the template we do not allow the grey value in any pixel to exceed the upper bound 0.005 and in the optimization of weights we do not allow any weight to exceed the same upper bound 0.005.

Optimization of the Weights and then the Template

We start with a certain template and weights and optimize the weights at first. The results have been already described in previous chapters. Then we use the two-stage approach to optimize the template for these optimal weights. The results are summarized in the top half of Table 3.12 in the sucessive steps: it is started with the initial worst separation, then the weights are optimized

(first the analytical and then the two-stage search) and finally the template is optimized with the optimal weights (again the analytical and then the two-stage search).

Using the bearded template and starting with radial weights, the solution of the analytical search (Figure 3.12 in the first row in the middle) improves the worst separation from 1.11 to 1.68 and the approximative search with $c = 0.005$ further to 1.73 (Figure 3.12 in the first row on the right). Then we take these resulting weights, start with the bearded template and apply the analytical search to optimize the template. The worst separation is improved only to about 1.75 and the resulting template is very similar to the original bearded template. Further the approximative search for the optimal template modifies the template only in several pixels as shown in Figure 3.28. The grey values of the template are namely reduced to zero in some pixels at its sides or the bottom. The improvement of the worst separation is however only slight and does not exceed 1.75, which can be also found in the first row of Table 3.12.

The initial worst separation with equal weights and the bearded template equals 0.78, the analytical search for optimal weights improves it to 1.67 and the two-stage search to 1.94. Using these weights (Figure 3.12 in the second row on the right) and starting with the initial bearded template, the analytical and then the approximative search for the optimal template has been applied. However the effect of optimizing the template is again negligible and does not bring any further improvements beyond 1.94.

Further we have performed the computations with three choices of random weights. These are the initial weights from the third, fourth and fifth row of Figure 3.12. The optimization of the template improves the worst separation only in some cases as shown in Table 3.12 and the resulting templates are visually very similar to the initial bearded template. The best result is obtained with the fifth set of initial weights (Figure 3.12 fifth row left) and the worst separation over 124 images attains 2.16.

Optimization of the Template and then the Weights

Now we start with a given template and weights, optimize the template by the two-stage search and only then the weights are optimized for this optimal template again using the two-stage search. The worst separation is shown in the bottom half of Table 3.12 starting with the initial situation, then for the optimal template using the initial weights and finally for the optimal template and optimal weights.

Let us start with radial weights and the bearded initial template. We recall that the worst separation equals 1.11. The optimal template obtained with the analytical search is shown in the middle of the first row of Figure 3.29 and the worst separation equals 1.79. The lips play the prominent role in the template, but the beard does not any more. The approximative search does not modify the template any further. Now we take this optimal template and apply the analytical search to optimize the weights starting with radial weights. The worst separation is improved to 1.83 and the weights are visually very similar to the radial ones. The approximative search improves the weights further. These are shown on the right of the first row of Figure 3.29 and yield the worst separation 1.90. These values are shown in the first row of the bottom half of Table 3.12.

Starting with the bearded template and equal weights, the worst separation over the whole database of images equals 0.78. Then we apply the analytical search to optimize the template. The result is shown in the middle of the second row of Figure 3.29 and its similarity to the template obtained with radial weights is remarkable. The worst separation equals 2.12. The approximative search does not modify the template further. Just like in the previous paragraph

the analytical search improves the template so much that the two-stage search is not necessary. This optimized template is now used together with equal weights and the analytical search is applied to optimize the weights, which improves the worst separation to 2.21. Finally the approximative search is applied on the weights. The resulting weights are shown on the right of the second row of Figure 3.29 and the worst separation is improved further to 2.29.

Figure 3.29 contains results also for three random choices of initial weights. The initial weights are shown in the left column. The optimized template in the middle column looks in each case very similarly. In the third row of the figure the optimization of the weights improves the worst separation only slightly. Starting with the initial weights of the fourth row the analytical search increases the worst separation to 2.21, the weights are similar to the initial ones and the approximative optimization further increases the worst separation by increasing the weights in some pixels up to 0.0032 and on the other hand reduces 7 % of the weights to zero. Finally with the last choice yields the very best results. The optimal template and the initial weights give the worst separation 2.19, the analytical search for optimal weights does not change much and the approximative search for optimal weights brings a further improvement to 2.30. We recall that no grey value of the template nor any weight exceeds the upper bound 0.005.

Comparison

Table 3.12 compares two different approaches, one of which optimizes the weights and then the template, and the other optimizes the template and then the weights. Comparing the top and bottom half of the table we can conclude that the best results are obtained by optimizing the template at first and then the weights.

Actually optimizing the template itself improves the worst separation even more than optimizing the weights followed by optimizing the template. The resulting optimized templates are very similar to each other independently on the choice of initial weights, but let us keep in mind that the same initial template was used. Furthermore the approximative search is not needed for optimizing the template, which indicates good convergence properties of the analytical method for approximating the optimal template.

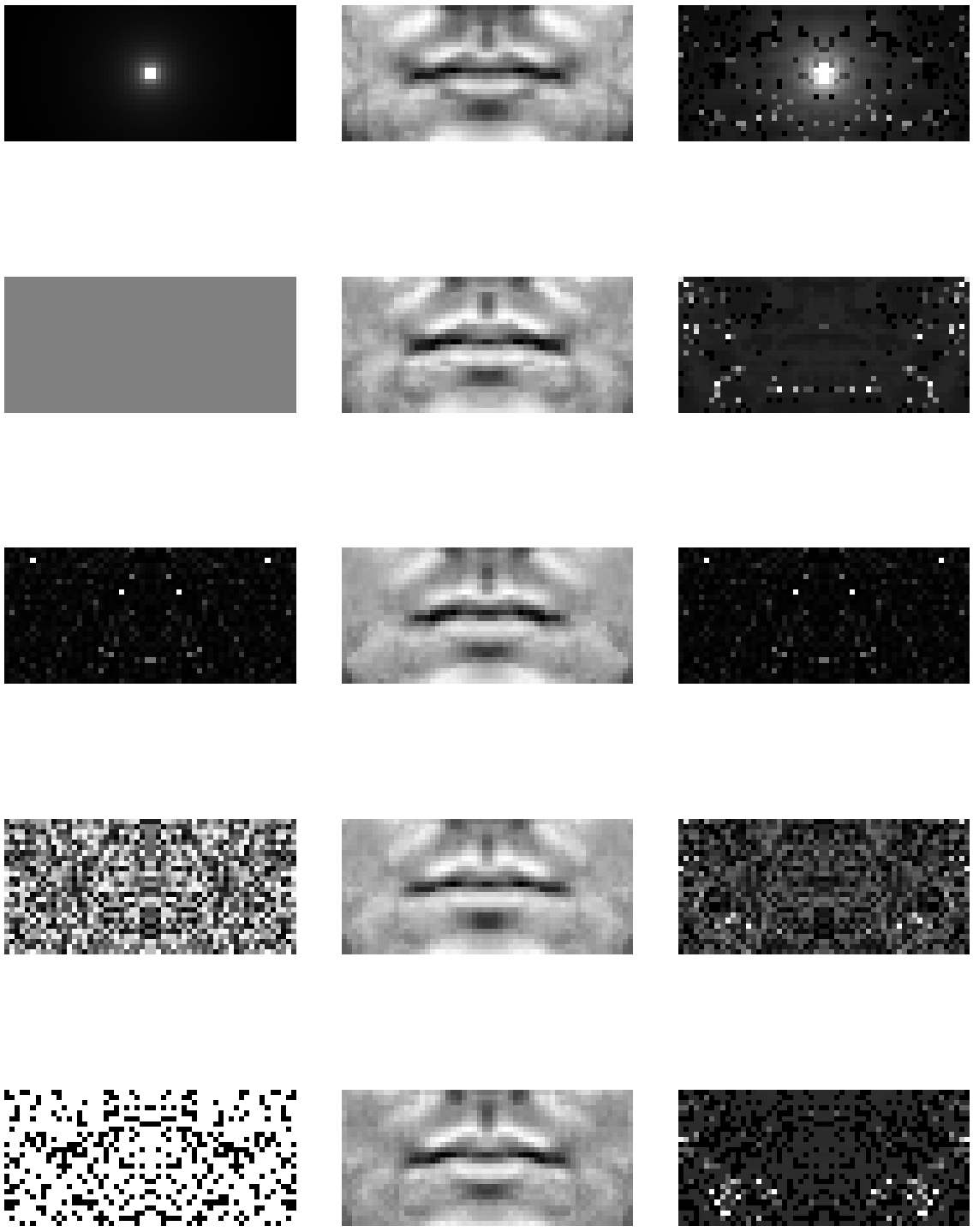

Figure 3.29: Optimal template obtained with different weights and bearded initial template. Left: initial weights. Middle: optimal template. Right: optimal weights for the optimal template. Results of the two-stage search with the upper bound $c = 0.005$.

Chapter 4

Locating Landmarks in Faces—Other Results

This chapter presents some other results in locating landmarks in faces independently on the methods of previous chapters. In Chapter 4.1 the search for the eyes using template matching is simplified by using an additional information about the position of the mouth. The mouth and eyes are jointly located in Chapter 4.2 and the methods turn out to be not only robust to a small rotation of the face, but yield correct results for any rotation of the face while rotating the templates by all possible angles. An algorithm for locating the nostrils is described in Chapter 4.3, which assumes the mouth and eyes to be already located. Finally we give some additional comments to the whole topic of this work and suggestions for a future research.

4.1 Locating the Eyes Using the Information about the Mouth

In Chapter 2.3 the construction of eye templates is described. The right and left eyes are located correctly in all images of the database with a set of six templates for the right eye and their mirror reflections. However the method turns out to be very sensitive to a different size or rotation of the face. Therefore we now assume that the position of the mouth is known and use this information in the search for both eyes. The template matching is used and each of the eye templates is placed to the horizontal strip shown in the left image of Figure 4.1. This starts 20 pixels above the mouth and its top boundary has the vertical distance from the mouth 55 pixels.

The results with the set of twelve eye templates from Chapter 2.3 are given in the top left part of Table 4.1 which otherwise contains other results described in later chapters. The position of the mouth is known and we use 12 eye templates to locate the eyes. The classical sample correlation coefficient used as the similarity measure locates the eyes correctly in all 124 images.

We use the weighted correlation coefficient with radial weights as the similarity measure between the template and the image. The eyes are located again in 100 % of the images. We note that the templates are created together with the classical correlation, namely as mean of eyes with a large correlation with initial eyes. Nevertheless they perform well with the weighted correlation too and the eyes are located in each of the 124 given images.

The limitation of suspicious areas shown in Figure 4.1 (left) is rather week and there could be a further reduction of the area in the horizontal direction, because the eyes never appear directly near the right or left edge of the image. However too many limitations could make the method too sensitive to different conditions such as different size or rotation of the face. The

Figure 4.1: Left: the eyes are searched for in the horizontal strip based on the known position of the mouth (Chapter 4.1). Right: the eyes are searched for based on a suspicious mouth (Chapter 4.2).

performance of the method for a different size or rotation is presented in Table 4.1, which will be described in the next chapter.

4.2 Mouth and Eyes Together

In this chapter the mouth and eyes are located together at the same time. The idea is to search for the eyes in a certain region above suspicious mouths.

We use seven mouth templates and six templates for the right eye described in Chapter 2.3 and their mirror reflections to find both eyes. Starting with the mouth templates we find not only one but several suspicious areas, which have the correlation with one of the mouth templates larger than a certain threshold. This is different in every image and is selected so that there are at least three such suspicious areas not directly neighbouring with each other. We start with 80 per cent of the maximal correlation which is attained. If necessary this threshold is decreased.

Certain suspicious areas correspond to each suspicious mouth. Figure 4.1 (right) shows one suspicious mouth and there are two rectangular regions above it, where the eyes are searched for. These starts 20 pixels above the suspicious mouth and their top boundary has the vertical distance from the mouth 55 pixels. The six eye templates and their mirror reflections are placed on every possible position in the two rectangles and the (weighted) correlation is computed between the templates and corresponding areas. We search for one eye in one of the rectangles and the other eye in the other rectangle. However the eye templates together with the mirror reflections are used in both rectangles. This is repeated for several suspicious mouths.

Moreover we use the restriction that the eyes are more than 25 and less than 42 pixels distant from each other. These bounds were found empirically to be fulfilled for eyes even in pictures with a small difference of the size of the face or a with its rotation by a small degree.

For a given suspicious mouth we find the suspicious eye in both of the two rectangles. If the condition on the distance of eyes is not fulfilled, then we search for an area different from these two suspicious eyes with the largest correlation among all remaining areas with the midpoint in one of the two rectangles. One suspicious eye is then this pixel and the other one is the previously

Table 4.1: Locating the mouth and eyes using methods of Chapters 4.1 and 4.2. Percentages of correct results.

Templates	Original		Smaller face		Rotated face	
	r	r_W	r	r_W	r	r_W
Locating the eyes with 6 templates using the information about the position of the mouth						
	1.00	1.00	0.80	0.97	0.50	0.86
Locating the mouth and eyes together; eyes searched for in a more restricted area:						
Mouth: 7 templates	1.00	1.00	0.99	1.00	1.00	1.00
Mouth: Bearded	0.98	1.00	0.80	0.99	0.90	0.98
Locating the mouth and eyes together; a more relaxed version:						
Mouth: 7 templates	1.00	1.00	0.99	0.99	0.92	0.87
Mouth: Bearded	0.98	1.00	0.80	0.99	0.90	0.98

suspicious area in the opposite rectangle. The condition is checked again and possibly the steps are repeated until suspicious areas with a suitable distance are located. The results are contained in Table 4.1. This approach considerably improves the results for rotated pictures, which will be discussed below.

The idea is to sum the three weighted correlations corresponding to the mouth and both eyes. Therefore we start with one of the pixels, which is in the midpoint of an area with a large weighted correlation with any of the mouth templates. Let us denote the largest of these weighted correlations between the area and any of the mouth templates by r_{W1}. We place all eye templates to all possible positions with the midpoint in the left part of the horizontal strip. Searching for the eyes only in the left part of that rectangle we obtain the largest weighted correlation with any of the eye templates which we denote by r_{W2}. In a similar way we place all the eye templates to the right part of the admissible horizontal strip and find the largest weighted correlation which will be denoted by r_{W3}. We consider the measure

$$(r_{W1} + r_{W2} + r_{W3}) \cdot \mathbb{1}(r_{W1} > 0) \cdot \mathbb{1}(r_{W2} > 0) \cdot \mathbb{1}(r_{W3} > 0), \tag{4.1}$$

which neglects negative weighted correlation. Then another suspicious mouth is taken and again the largest possible value of (4.1) found.

The first row of Table 4.1 was described in the previous section. The remaining rows describe results of this section, namely the percentage of correct locating of the mouth and eyes together. The first two columns apply to standard images using the classical correlation and the weighted correlation with radial weights. For the mouth we use either all seven templates or only one with the beard. For the eyes we use six templates for the right eye and also their mirror reflection.

To show the importance of the restriction on the horizontal distance of the eyes, we have performed the search for the landmarks also without it. The results are weaker as shown in the bottom part of Table 4.1. Later in Chapter 4.3 the more restrictive method turns out not to be more sensitive to the size of the head or its rotations, but actually performs better in correct locating the mouth and eyes under these different conditions.

Figure 4.2: Mouth and eyes in a picture rotated by −10 degrees.

The Exact Position of the Mouth

When the mouth and eyes are located, we use an additional procedure to specify more exactly the position of the midpoint of the mouth. Template matching finds namely the suspicious area including the mouth, which does not necessarily have the midpoint of the lips in its midpoint. The midpoint of the mouth is not precisely in the midpoint of every template and it may be necessary to search for the exact position of the midpoint of the mouth as well. The following simple approach is successful thanks to the fact that the mouths in our database are not affected by facial expressions.

The horizontal line between the lips has lower (darker) grey values than the rest of the mouth with its neighbourhood. First we search for just one pixel of the line, then the whole line itself and finally the midpoint of the mouth is the midpoint of this line. A simple approach is to find the darkest pixel in the neighbourhood of that pixel, which is the midpoint of the area with the largest (weighted) correlation with the template. If the largest weighted correlation is obtained with one of the bearded templates, the lips should be searched for in the top part of the suspicious rectangle. Each of the bearded templates has the mouth in its top part. Thus we locate one pixel of the horizontal line of the mouth in each picture correctly.

The whole horizontal segment of the mouth can be reliably localized only after locating the eyes and using the information about their position. If the eyes are horizontal, the midpoint of the mouth is located straight below the point in the middle between eyes assuming the symmetry of the mouth. Therefore it is no disadvantage that the mouth template itself does not locate the midpoint of the mouth precisely and it is possible to consider the result of the search to be successful also when a pixel in the neighbourhood of the midpoint is located, just like in Figure 3.1.

Figure 4.2 shows the result for a picture rotated by −10 degrees. The mouth and eyes have been found using the method of this chapter. The eyes do not have a horizontal position so the segment of the mouth is not estimated to be horizontal but rather rotated by the angle determined by the position of the eyes. The midpoint of the mouth can be found as the midpoint of that line segment.

Different Size or Rotation

We examine the robustness of the methods to the size and rotation of the faces. In Table 4.1 there are results of locating the eyes using Chapter 4.1; the combined search for the mouth and eyes using Chapter 4.2; and finally results of a more relaxed version without the restriction on the horizontal distance of the eyes.

First we examine the rotation to the *size*. When the position of the mouth is known, the performance of locating eyes is worse than in the original images. Neither the sample correlation nor the weighted correlation with radial weights locate the eyes in every image.

Locating the eyes with mouth together in pictures smaller by 10 % is successful in every image with the stricter approach when the distance between the eyes is limited and when the radial weights are used. We recall that all the seven templates locate the mouth in the smaller images in every image, but the bearded template itself does not. This was presented in Table 2.2. There the bearded template is denoted by the reference number 6.

Locating the mouth with seven mouth templates and eyes with six templates and a restriction on their distance locates the landmarks correctly in the entire database and also when the size of the face is smaller by 10 %.

Now we give results of robustness to the *rotation* of the face, while the templates themselves are not rotated. When the position of the mouth is known, locating the eyes does not perform well in faces rotated by ± 10 degrees.

We recall that no method in previous chapters was able to locate the mouth in every image rotated by ± 10 degrees (Table 2.2). Now while locating the mouth and both eyes together using the more restricted search it turns out that one method has the 100 % performance in locating all these three landmarks. This method uses six eye templates together with their mirror reflections and seven mouth templates and the radial weights for the weighted correlation coefficient. The same eye templates and only the bearded mouth template are too sensitive, mainly because the mouth template itself is too sensitive to rotation. The less restricted approach without the condition on the horizontal distance of eyes is too sensitive to rotation.

Rotating the faces by a larger rotation of ± 20 degrees however makes the method fail in about one half of images. Then the weighted correlation of a rotated mouth with nonrotated mouth templates falls as far downwards as to 0.4, while other areas such as the chin, hair or eyebrows still can have the weighted correlation with a mouth template over 0.7.

If the face is rotated, the sum of the three weighted correlations is lower than for the face in the upright position. A good strategy is to rotate the face by several different angles and for each possibility to find the maximal sum of the three weighted correlation in the picture. This method locates the mouth and eyes correctly independently on the initial rotation of the given image. This is confirmed in the next text.

We have rotated each picture by angles of 0, 10, 20, ..., 180, ..., 350 degrees. In all of the 124 images, the largest value of the separation measure (4.1) was attained for the nonrotated picture. As described above, the mouth and eyes are located correctly in all of the 124 images.

To summarize, the described method of locating the mouth and the eyes is *twice* protected against a possible rotation of the face. Firstly the templates are robust to rotation up to ± 10 degrees. And secondly the loss function (4.1) is the largest exactly for the nonrotated face. The correct rotation of the face is thus found for every picture with any rotation up to ± 180 degrees.

Now we have the solution to the problem of analyzing images of faces rotated by a small

degree. In Chapter 1.1 we have explained that such images appear in the database coming from the Institute of Human Genetics and that estimating this small rotation is of a key interest. The method of this chapter allows to rotate automatically the images to have the eyes horizontally.

In a given face we locate the mouth and eyes together (using methods of Chapter 4.2) correctly even in spite of a possible rotation of the face. The darkest point of the pupil is located using the method of last paragraph of Chapter 2.3. We propose to describe the rotation of the face by the position of eyes and rotate the image to bring the eyes to the horizontal position. Then the search for remaining landmarks becomes independent on the initial rotation.

After determining the rotation in images by our procedure the commercial software can be applied to search for other landmarks. This can be for example the procedure sensitive to rotation, which is described in Chapter 1.1.

Results in the New Database of Images

The combined search for the mouth and eyes uses seven mouth templates and six eye templates together with the condition on the distance of eyes. It locates the mouth and both eyes in every image in the database containing 124 images and is robust to changes of the size by 10 % and robust to rotation by ± 10 degrees.

We have applied the joint search of the mouth and both eyes in another database of 88 images, which have been photographed at the Institute of Human Genetics under the same conditions as the original images. The mouth was classified well in 99 % of images and both eyes are found correctly in 99 % of images. Next we have modified the algorithm to make it working for all of the 124+88=212 images. Such improvement required several changes and these will be now described for the mouth and then for the eyes.

The combined search for mouth and eyes fails in locating the *mouth* in one of the new images, namely in an image of a man without beard. Adding his mouth to the set of mouth templates makes the method fail for another image, namely a lady in the original database. The mean of the mouths of these two faces added to the set of templates causes the method to fail only in one image, which is a bearded man from the new database.

The mean of the three problematic mouths does not help in locating the mouth. Therefore we have focused on the bearded mouths. There are two bearded templates in our set of seven mouth templates. We keep one of them (Figure 3.2 left) as a template. We compute the mean of the other bearded template and the bearded mouth of the man from the last sentence of the last paragraph. The second bearded template is then replaced by this new template.

The method then locates the mouth correctly in all 212 images. To summarize, one of the bearded templates was replaced and another nonbearded was added, so that there are eight mouth templates altogether.

The combined search for eyes and mouth uses six templates for the *eyes*. It fails in correct locating the eyes in one of the new images, where the eyes are too far apart to fulfill the condition on their distance. We change this condition and introduce a new eye template. The eyes must now lie more than 25 and less than 45 pixels far apart. The upper bound is now more liberal than before. Then the eyes are located correctly in the whole new database, but in one of the original images the hair becomes suddenly suspicious. The problematic eye of that person becomes a new template. The condition on the horizontal distance of the eyes is especially helpful in images with a different size or rotation of the face. Without the condition the eyes

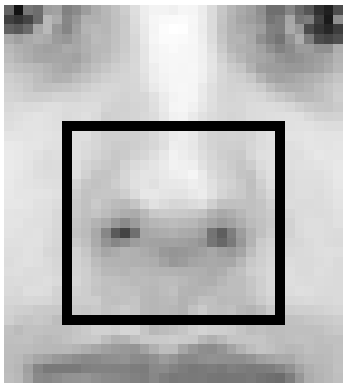

Figure 4.3: Search for the nostrils.

are actually located correctly in 99 % of standard images.

The improved method uses eight mouth templates and seven eye templates. The mouth and both eyes are located correctly in all of the total number of 212 images from both databases and also in 100 % of images with a face smaller by 10 %. The mouth and eyes are also located correctly in 100 % of images rotated by $+10$ or -10 degrees. Further we describe the robustness of the method to a different size or rotation of the face.

4.3 Locating the Nose

We assume the mouth and eyes to be already located. Based on previous chapters we can now assume the midpoints of both eyes to be estimated precisely and the mouth segment between the lips as well. We rotate the image to bring the eyes to a horizontal position while not losing the information about the position of the mouth and eyes themselves. This information can be now used in the search for the nostrils, which is relatively easier compared to locating the mouth and eyes themselves. Figure 4.3 gives an illustration. The idea is that the nostrils are the two darkest areas in the rectangle between the mouth and the eyes.

The simplest approach is to search for the areas with the smallest grey values between the mouth and the eyes. However we do not consider the whole rectangle, but only its inner part shown in Figure 4.3 inside of the black boundary. We cut off 6 pixels from the left, right and bottom and 12 pixels from the top. A naive idea is to count the number of areas in the inner part of the rectangle which are darker than a certain threshold. However there does not exist one suitable threshold uniformly over all images which would discriminate the nostrils from the rest of the rectangle in every image. Therefore we search for a suitable threshold separately in each image. In this inner part of the rectangle we count the number of different areas darker than a very low threshold. Then we repeatedly increase the threshold until exactly two areas are located, which are not direct neighbours of each other.

This approach does not locate the nostrils correctly for such pictures, in which one of the nostrils is indistinct. Our database namely contains 4 % of pictures with head rotated downwards or to the side. There is one of the nostrils is covered by the nose and is not visible in such images. We carry out the following procedure to identify such problematic pictures automatically.

We compare the Euclidean distance of the two darkest areas and the grey values in them.

We find thresholds in such a way that the method works also in images smaller by 10 % and is at the same time robust to rotating the face up to 10 degrees in either direction. Also the size of the inner part of the rectangle (Figure 4.3) was actually selected to be robust to the size of the face.

Then we do *not* consider the two darkest areas in the inner part of the rectangle to be the nostrils, if the two conditions

$$(\text{distance} \leq 9 \text{ pixels}) \quad \text{and} \quad (\text{grey values} > 0.27) \tag{4.2}$$

are fulfilled at the same time. In other words, we doubt about such two suspicious areas being the nostrils, which are too close to each other and at the same time too light. Both conditions of (4.2) are robust to small changes of the size of pictures.

When the condition (4.2) is fulfilled and the two suspicious areas cannot be trusted to be the nostrils, we use an alternative approach. In this case it turns out that the two darkest areas in the inner part of the rectangle always include one of the two nostrils and the bottom part of the nose, so the position of the other nostril can be already well approximated.

In one of our pictures it happens however that the second and third darkest areas have the same grey values and there are three rather than two darkest areas. Such situation is identified by our program automatically. In this case the areas most on the right and left are the nostrils and the middle one corresponds to the bottom of the nose.

The nostrils are located correctly in every picture. If one of them is missing, the program recognizes this automatically and estimates its location reasonably well in every image. These results remain valid for pictures with a size smaller by 10 %. The robustness against rotation is ensured, because rotating the face to have the eyes horizontally is carried out before searching for the nostrils.

Another approach is to estimate the position of the nostrils based on their usual position between the eyes and the mouth, for example the mean position. We have however observed a large variability in the position of nostrils in the database of images.

The vertical distance between the eyes and mouth varies between 27 and 46 pixels. The vertical distance between the eyes and nostrils attains more than 52 % and less than 71 % of the distance between the eyes and the mouth. In this region the nostrils can be expected. There is a similar variability in the horizontal distance between the nostrils. The horizontal distance between the left iris and the left nostril happens to be between 23 % and 38 % of the distance between the eyes. The horizontal distance between the right iris and the right nostril lies between 27 % and 47 % of the distance between the eyes.

Estimating the position of the nostrils based only for example on their mean position is therefore very inaccurate. They can be however searched for in a smaller inner part of the rectangle between the eyes and nose or it would be possible to combine the information about the expected position of the nostrils with the search for the two darkest areas.

We have applied the algorithm searching for the nostrils as two darkest areas also to the other database of 88 new images. Then the nostrils are located correctly in 96 % of cases. Now we propose the following classification tree which finds the nostrils correctly in the entire set of 212 original and new images. This searches again for two darkest areas but estimates the nostrils by their mean positions in a small percentage of problematic cases. We do not try to make the method robust to rotation because we assume that the eyes are in a perfectly horizontal position.

We consider the inner part of the rectangle just like described above. First we try to find the two darkest areas. In 1 % of images the second and third darkest area have the same grey values. In that case the very darkest area is classified as a nostril and the position of the other nostril is then estimated using symmetry.

Let us now assume that the two darkest areas in the inner part of the rectangle have been located. Then we compare their distance and grey values with the following thresholds.

If the distance is less or equal to 9 pixels and at least one of the nostrils does not contain grey values lower than 0.45, then the two suspicious areas are too light and too close to each other to be classified as nostrils. Then we estimate their positions by the mean positions of nostrils computed from all images. This happens in 2 % of images. Compared with the previous chapter, the thresholds were changed in order to work also for the new 88 images.

If the distance is more than 16 pixels and at least one of the nostrils does not contain grey values lower than 0.4, then the two suspicious areas are too light and too far apart to be classified as nostrils. Then we again estimate their positions. This happens in 1 % of images.

If none of the conditions of the two last paragraphs is fulfilled, then we classify the two suspicious areas as nostrils.

This approach located nostrils well or at least approximates their positions reliably in all 212 images. For the problematic cases we have observed that in each of the first problematic case (nostrils too light and too close to each other) the very darkest area is a nostril. In the second case (nostrils too light and too far apart) the darkest area is either the bottom of the nose or a shadow next to the side part of the nose. Assuming this, the correct nostril could be retained and the other one could be estimated using the symmetry. However such decision rule relies too heavily on special properties of the given images. Therefore we prefer the rough estimation of the positions of the nostrils by the mean. This happens in only 3 % of images and turns out to give reasonable results.

There are also other possible approaches to locating either one of the nostrils or both of them at the same time, which we have not implemented:

- Search for areas with a large total variation. This will be large at the boundary between the black nostrils and their light grey neighbourhood.
- Use a template for a nostril which has a black middle and lighter neighbourhood.
- Use a template for both nostrils together with two small black circles inside a lighter area.
- Examine grey values over horizontal lines through the square in Figure 4.3. There are two deep valleys corresponding to the two black nostrils.
- Examine circular areas with a fixed radius of the rectangle of Figure 4.3. In every possible circle count the pixels which are darker than a certain threshold. The largest number of such pixels is in the nostrils.

4.4 Final Remarks. Future Research.

The whole thesis describes methods for locating landmarks in images of faces using templates. The optimization of weights for the weighted correlation coefficient presented in Chapter 3 is a general method and can be interpreted as a method for robust nonparametric discrimination.

It has been the aim of this research to examine the possibility of using templates to locate the landmarks, the strength and weaknesses of templates. Therefore the methods of this work use as little prior information as necessary. For example we search for the mouth in the whole image, although it appears always in its bottom half. A similar additional information could be used in the search for the eyes. Such restrictions would make the task easier, but on the other hand sensitive to extreme rotations of the face. Another simplification is to start the image analysis by separating the outline of the person from the background using boundary extraction techniques (Jain 1989). Then the landmarks can be search for only within the body area.

Another topic of the research has been the search for the vertical axis of the face in images, which is not presented in this thesis. There are two rectangular areas compared, namely on the left and on the right side of each vertical line in the image. Similarity between one rectangle and the mirror reflection of the other one is measured. In that context we have compared different correlation measures including robust ones and the weighted correlation coefficient with simple weights (similar to radial) can be recommended. The search for the vertical axis of the face suffers however from a sensitivity to rotation.

Finally we give some suggestions for a future research. One possibility is to search for remaining landmarks. The position of the mouth and eyes can be now assumed to be known and that allows to estimate the rotation and size of the face. Therefore the mouth and eyes play a crucial role in locating all the landmarks and we believe that the remaining ones can be located in a way which is relatively less complicated.

Another task is the automatic detection of special effects such as glasses. The faces in the database from the Institute of Human Genetics do not contain glasses, but we have observed in other images that the glass sometimes reflects light and the eyes cannot be well seen. But even if this is not the case, the rim of the glasses counteracts in the neighbourhood of the eye as a part of the area compared with the template. Another difficulty is a 3D-rotation of the face (rotation in pose), which has not been systematically analyzed in literature for the template matching yet.

A practical task is to combine the methods of Chapters 3 and 4 to search for optimal weights in a joint search for the mouth and both eyes together.

Other measures of correlation should be more systematically examined, for example robust analogies of the correlation coefficient such as various versions of a trimmed correlation coefficient or a weighted correlation with weights depending on the residuals of the linear regression fit. A first step would of such work would require to propose a fast approximative algorithm and to study its computational aspects.

In Chapter 3.7 a transformation is applied on the data, which is motivated by the autocorrelation structure of the images. It is imaginable that other preliminary transformations may be suitable either to improve the worst separation even further or to reduce the dimension of the computation and increase its speed. We have not made systematic effort to examine such possible transformations. One such possibility is also a preliminary denoising of the image for example by means of a two-dimensional filter based on robust regression with a high breakdown point.

A final application may be a general subroutine for locating landmarks in a general context. That would create the templates, suggest suitable initial weights and optimize both the template and the weights automatically.

Bibliography

[1] Amit Y., Grenander U., Piccioni M. (1991): Structural image restoration through deformable templates. *J. Amer. Statist. Assoc.* **86**, No. 414, 376 – 387.

[2] Belhumeur P.N., Hespanha J.P., Kriegman D.J. (1997): Eigenfaces vs. Fisherfaces: Recognition using class specific linear projection. *IEEE Trans. Pattern Anal. and Machine Intel.* **19**, No. 7, 711 – 720.

[3] Böhringer S., Vollmar T., Tasse C., Würtz R.P., Gillessen-Kaesbach G., Horsthemke B., Wieczorek D. (2005): Syndrome identification based on 2D analysis software. Submitted.

[4] Bronstein A.M., Bronstein M.M., Kimmel R. (2005): Three-dimensional face recognition. *Int. Journal of Comp. Vision* **64**, No. 1, 5 – 30.

[5] Brunelli R., Poggio T. (1992): HyperBF networks for gender classification. *Proceedings DARPA Image Understanding Workshop*, San Diego, 311 – 314.

[6] Davies P.L., Gather U. (2004): Robust Statistics. In Gentle J.E., Härdle W., Mori Y. (eds.): Handbook of computational statistics. Concepts and Methods. Springer, Heidelberg, 655 – 695.

[7] Dobeš M., Machala L., Tichavský P., Pospíšil J. (2004): Human eye iris recognition using the mutual information. *Optik* **115**, No. 9, 399 – 404.

[8] Downie T.R., Shepstone L., Silverman B.W. (1996): A wavelet based approach to deformable templates. In Mardia K.V., Gill C.A., Dryden I.L. (eds.): *Proceedings in Image Fusion and Shape Variability Techniques*, Leeds University Press, Leeds, 163 – 169.

[9] Downie T.R., Silverman B.W. (2001): A wavelet mixture approach to the estimation of image deformation function. *Sankhya* **43**, Ser. B, Pt. 2, 149 – 166.

[10] Er M.J., Chen W., Wu S. (2005): High-speed face recognition based od discrete cosine transform and RBF neural networks. *IEEE Trans. Neur. Networks* **16**, No. 3, 679 – 689.

[11] Graf H.P., Cosatto E., Gibbon D., Kocheisen M., Petajan E. (1996): Multi-modal system for locating heads and faces. *Proc. Second Int. Conf. Automatic Face & Gesture Recognition*, IEEE Computer Soc. Press, Los Alamitos, 88 – 93.

[12] Greene W.H. (1993): Econometric analysis. MacMillan, New York.

[13] Grenander U. (1993): General pattern theory. A mathematical study of regular structures. Oxford University Press, Oxford.

[14] Hammond P., Hutton T.J., Allanson J.E., Campbell L.E., Hennekam R.C.M., Holden S., Patton M.A., Shaw A., Temple I.K., Trotter M., Murphy K.C., Winter R.M. (2004): 3D analysis of facial morphology. *Amer. J. Med. Genet.* **126** A, 339 – 348.

[15] Hancock P.J.B. (2000): Evolving faces from principal components. *Behav. Res. Methods Instrum. Comput.* **32**, No. 2, 327 – 333.

[16] Hancock P.J.B. (2005): Principal components of shape variation of faces. Animations. `www.stir.ac.uk/psychology/Staff/pjbh1/facepca.html`.

[17] Hastie T., Tibshirani R., Friedman J. (2001): The elements of statistical learning. Springer, New York.

[18] Hays W.L. (1973): Statistics in the social sciences. Holt, Reinhart and Winston, London.

[19] Huang J., Blanz V., Heisele B. (2002): Face recognition with support vector machines and 3D head models. *Proceedings of the International Workshop on Pattern Recognition with Support Vector Machines (SVM 2002)*, Niagara Falls, 334 – 341.

[20] Jain A.K. (1989): Fundamentals of digital image processing. Prentice-Hall, Englewood Cliffs.

[21] James M. (1987): Pattern recognition. BSP Professional books, Oxford.

[22] Kanters F., Lillholm M., Duits R., Jansen B., Platel B., Florack L., ter Haar Romeny B. (2005): On image reconstruction from multiscale top points. *Lecture Notes in Computer Science* **3459**, 431 – 439.

[23] Lindeberg T. (1994): Scale-space theory in computer vision. Kluwer, Dordrecht.

[24] Loos H.S., Wieczorek D., Würtz R.P., Malsburg von der C., Horsthemke B. (2003): Computer-based recognition of dysmorphic faces. *Eur. J. Hum. Genet.* **11**, 555 – 560.

[25] Moran P. A. P. (1950): Notes on continuous stochastic phenomena. *Biometrika* **37**, 243 – 251.

[26] Osuna E., Freund R., Girosi F. (1997): Training support vector machines: An application to face detection. *Proceedings CVPR 1997 (IEEE Computer Society Conference on Computer Vision and Pattern Recognition)*, IEEE Computer society press, Los Alamitos, 130 – 136.

[27] Press W.H., Teukolsky S.A., Vetterling W.T., Flannery B.P. (1992): Numerical recipes in Fortran 77. The art of scientific computing. Cambridge University Press, Cambridge.

[28] Ramsay J.O., Silverman B.W. (2002): Applied functional data analysis. Springer, New York.

[29] Rencher A.C. (1998): Multivariate statistical inference and applications. Wiley, New York.

[30] Rowley H., Baluja S., Kanade S. (1998*a*): Neural network-based face detection. *IEEE Trans. Pattern Anal. and Machine Intel.* **20**, No. 1, 23 – 38.

[31] Rowley H., Baluja S., Kanade S. (1998*b*): Rotation invariant neural network-based face detection. *Proceedings CVPR 1998 (IEEE Computer Society Conference on Computer Vision and Pattern Recognition)*, IEEE Computer society press, Los Alamitos, 38 – 44.

[32] Shevlyakov G.L., Vilchevski N.O. (2001): Robustness in data analysis: criteria and methods. VSP, Utrecht.

[33] Smith L. (2001): An introduction to neural networks.
`http://www.cs.stir.ac.uk/~lss/NNIntro/InvSlides.html`.

[34] Starck J.-L., Murtagh F., Bijaoui A. (1998). Image processing and data analysis. The multiscale approach. Cambridge University Press, Cambridge.

[35] Stergiou C., Siganos D. (1996): Neural networks. *SURPRISE 96 Journal*, Vol. 4, Imperial college of science, technology and medicine, London.

[36] Sultan A. (1993): Linear programming. An introduction with applications. Academic Press, Boston.

[37] Vapnik V.N. (1995): The nature of statistical learning theory. Springer, New York.

[38] Venables W.N., Ripley B.D. (1994): Modern applied statistics with S-Plus. Springer, New York.

[39] Winkler G. (1995). Image analysis, random fields and dynamic Monte Carlo methods. A mathematical introduction. Springer, Berlin.

[40] Wiskott L., Fellous J.-M., Krüger N., Malsburg von der C. (1997): Face recognition by elastic bunch graph matching. *IEEE Trans. Pattern Anal. and Machine Intel.* **19**, No. 7, 775 – 779.

[41] Wu J., Trivedi M.M. (2004): A binary tree for probability learning in eye detection. Computer vision and robotics research lab technical report, La Jolla.

[42] Würtz R.P. (1997): Object recognition robust under translations, deformations, and changes in background. *IEEE Trans. Pattern Anal. and Machine Intel.* **19**, No. 7, 769 – 775.

[43] Yang M.-H., Kriegman D.J., Ahuja N. (2002): Detecting faces in images: A survey. *IEEE Trans. Pattern Anal. and Machine Intel.* **24**, No. 1, 34 – 58.

[44] Yuille A. L., Hallinan P.W., Cohen D.S. (1992): Feature extraction from faces using deformable templates. *Int. Journal of Comp. Vision* **8**, No. 2, 99 – 111.