Robust Facial Component Detection for Face Alignment Applications

Martin Urschler[1], Markus Storer[1], Horst Bischof[1], Josef A. Birchbauer[2]

[1] Institute for Computer Graphics and Vision
Graz University of Technology, Austria
{urschler,storer,bischof}@icg.tugraz.at

[2] Siemens Biometrics Center
Siemens IT Solutions and Services, Siemens Austria
josef-alois.birchbauer@siemens.com

Abstract
Biometric applications like face recognition/verification are an important topic in computer vision, both for research and commercial systems. Unfortunately, state of the art face recognition systems are not able to work with arbitrary input images taken under different imaging conditions or showing occlusions and/or variations in expression or pose. To support face recognition one has to perform a face image alignment (normalization) step that takes occlusions/variations into account. In this work we present a robust face normalization algorithm suitable for arbitrary input images containing a face. The algorithm is based on detecting face and facial component candidates and robustly voting for the best face and eyes. Our restrictions are a certain pose range (frontal to half profile) and suitable illumination conditions. Our algorithm is designed to deal with occlusions and its performance is shown on three publicly available image databases.

1 Introduction

Face recognition is an important biometric application which is reflected in the relevance of facial image analysis as a research topic in computer vision. There is a high interest in biometrics due to a large number of potential commercial and law enforcement applications (e.g., biometric authentication or surveillance) requiring robust and highly accurate recognition of biometric features like fingerprints, iris or the face. The benefit of faces in this context is the lack of necessary cooperation and/or knowledge of the person as opposed to the other biometric cues. This topic has received significant research attention over the last two decades [23] and has also led to commercial systems [16].

We are specifically interested in analyzing facial portrait and near-portrait images in the context of the International Civil Aviation Organization (ICAO) standard [9] for machine readable travel documents (MRTDs). The main intention of this standard is to define how arbitrary images have to be prepared in order to perform robust and highly accurate face recognition/verification. For the purpose of geometrical alignment a part of the ICAO specification describes a standardized coordinate frame based on eye locations. A definition of this geometrical coordinate system forming normalized face portrait images (token images) is shown in Figure 1. This illustration clearly motivates the need to localize faces and facial components like eyes or mouth in arbitrary input images. Localization algo-

rithms suffer from problems due to occlusions like glasses, beards or covering objects (hands or hair) or facial expressions like an open mouth, closed eyes or facial grimaces. Images with differences in facial pose and illumination conditions further contribute to the difficulty of facial image analysis and normalization.

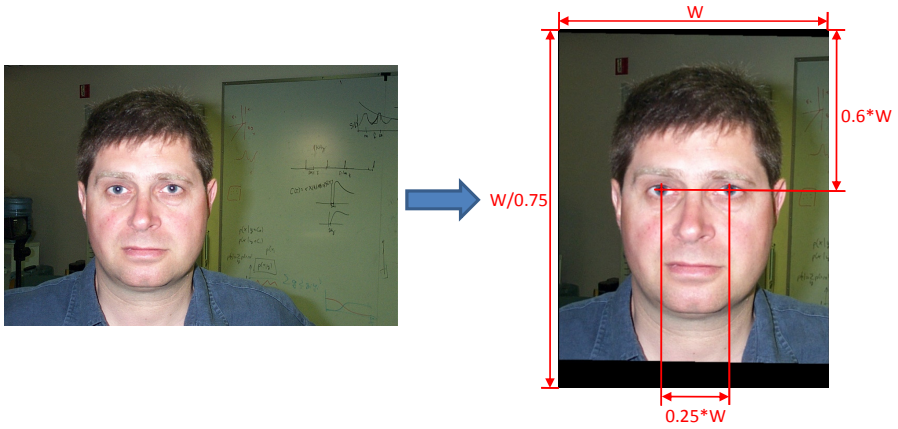


Figure 1: Illustration of the geometrical alignment procedure to form a normalized face portrait image (right) given an arbitrary input image (left). Image taken from the Caltech Faces database [1].

In this work we describe a robust facial image detection and normalization algorithm that may serve as a powerful pre-processing step for face recognition [23], face analysis and facial expression recognition [15]. Our method is based on separate modules for facial component detection which are combined in a probabilistic voting scheme to fuse the results of independent detectors, thus achieving robustness to occlusions and disturbances. The voting scheme results in face and eye locations that are used to calculate a face normalization transformation. The remainder of this paper is structured as follows. Section 2 presents the related work and Section 3 describes the components of our algorithm and the overall system. To demonstrate the strength of our approach we evaluate our algorithm on publicly available databases and compare it to state of the art methods in Section 4. Finally, we discuss our findings and conclude our work in Section 5.

2 Related Work

Detection of faces and facial components has a long tradition in computer vision research. Application areas dealing with face analysis/processing require an initial face localization step, see [22, 7] for literature surveys on face detection methods. Face detection from single images is challenging due to possible variabilities in scale, location, orientation, and pose. Further, different facial expressions, occlusions and illumination conditions also have an effect on the appearance of faces. According to [22], we define face detection as: Given an arbitrary image, the goal of face detection is to determine whether or not there are any faces in the image and, if present, return the image location and extent of each face. Further, facial feature detection has the goal of detecting facial components like eyes, mouth, nose, ears, etc., either under the assumption that there is only a single face in the image or given a known face detection.

In the face detection literature one distinguishes holistic (whole face is detected at once) and component based (non-holistic) approaches. Moghaddam and Pentland [13] describe a holistic principal component analysis (PCA) based detection system, where a PCA subspace (or eigenface representation) is used and the detection is performed in a sliding window manner. Rowley et al. [19] show a

neural network based approach for face detection, where the sliding windows are pre-processed and reduced in their dimensionality in order to be put into a neural network. Jesorsky et al. [10] describe a face detection method based on point sets and the Hausdorff distance. Schneiderman and Kanade [20] use multi-resolution information based on wavelet analysis to construct a nonlinear face and non-face classifier based on histogram statistics and the AdaBoost learning technique. Their approach inspired the important work of Viola and Jones [21] using simple and efficient Haar features which had an enormous impact on object detection over the last years. An example for a non-holistic approach is Heisele et al. [6] who exploit the idea of using components like mouth, eyes and nose for face detection. These components are related by constraints on their spatial configuration, which obviously makes the algorithm more robust to occlusions. A similar direction is proposed by Felzenszwalb and Huttenlocher [4] in their work on pictorial structures. A recent work in this direction is Erukhimov and Lee [3] who combine facial components with a graphical model. Our presented approach also uses the strategy to combine facial components for face detection and eye localization to be more robust against occlusions. A slightly different research direction for face detection makes use of statistical models of shape and/or appearance to localize faces and facial components, with the seminal work by Cootes et al. [2].

In the following paragraphs we describe two important basic components of our system in more detail. These are the efficient object detection approach from Viola and Jones [21] and the statistical shape and appearance model of Cootes et al. [2].

2.1 Efficient AdaBoost Object Detection

The most influential work in object detection of the last few years is definitely the approach of Viola and Jones [21] based on boosting simple weak classifiers to form a strong classifier. One of their application areas was face detection. Since we make use of their algorithmic framework in our face normalization system, we describe their method in this paragraph.

In machine learning AdaBoost [5] is a supervised classification technique to establish a complex nonlinear strong classifier

$$H_M(\mathbf{x}) = \frac{\sum_{m=1}^{M} \alpha_m h_m(\mathbf{x})}{\sum_{m=1}^{M} \alpha_m}$$

where $\mathbf{x}$ is a pattern to be classified, $h_m(\mathbf{x}) \in \{-1, +1\}$ are the M easily constructible, weak classifiers, $\alpha_m \geq 0$ are the combining coefficients, and $\sum_{m=1}^{M} \alpha_m$ is a normalizing factor. $H_M(\mathbf{x})$ is real-valued, however classification is done using the signum function $y(\mathbf{x}) = \text{sign}[H_M(\mathbf{x})]$.

The AdaBoost learning algorithm establishes a sequence of best weak classifiers $h_m(\mathbf{x})$ and their corresponding weights α_m. Confronted with N training examples of the form $\{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_N, y_N)\}$, where $y_i \in \{-1, +1\}$ are the class labels, a distribution of the training examples is calculated and updated during learning. After each iteration, examples which are harder to separate are investigated to put more emphasis on these examples. It is sufficient that weak classifiers are able to separate a training set better than simple guessing, i.e. it needs a classification rate larger than 50%. The trained weak classifiers determine the features that have to be evaluated in a new image.

In the original object detection framework [21] the weak classifiers are formed from the thresholded responses of simple Haar wavelet-like features. By calculating Haar features using a so-called integral image representation, it is possible to achieve high evaluation efficiency. The efficiency comes from the large number of features that are combined by the boosting approach and from the necessity to

evaluate features at different scales and locations. Another important advantage of the Viola and Jones approach is their cascaded structure of multiple strong classifiers, which is ideally suited for sliding window processing in object detection. Earlier cascade stages remove a large number of non-object windows, while it is necessary to traverse the whole complex cascade only for a window of the object class.

2.2 Localization Refinement based on Statistical Active Appearance Models

Generative model-based segmentation approaches for highly accurate feature localization have received a lot of attention in computer vision over the last decade. While the Viola and Jones approach solely leads to a coarse feature localization due to the sliding window approach, an exact object delineation is often performed using statistical models of shape and appearance. The most important representative of this class of algorithms is the Active Appearance Model (AAM) from Cootes et al. [2]. It has successfully found a large number of applications ranging from face detection to medical image analysis.

AAM describes the variation in shape and texture of a training set representing an object. From a mean shape and a mean texture, defined in the coordinate frame of the mean shape, the modes of shape and appearance variation are calculated by applying principal component analysis (PCA) to the geometrically and photometrically aligned training examples. Training examples are manually annotated with respect to their shape (commonly defined as corresponding landmark points) and are required to cover the types of variation present in the images one wants to analyze. Formally, from shape representations $\mathbf{x}$ and texture representations $\mathbf{g}$ a statistical model is built given N training examples (i.e., tuples $(\mathbf{x}_n, \mathbf{g}_n)$). The statistical model is based on the interpretation of this representation as a high-dimensional feature vector. A dimensionality reduction is performed by applying PCA to generate the more compact model

$$\mathbf{x} \approx \overline{\mathbf{x}} + \mathbf{P}_s \mathbf{b}_s$$

$$\mathbf{g} \approx \overline{\mathbf{g}} + \mathbf{P}_t \mathbf{b}_t$$

where shapes and textures are represented by the means $\overline{\mathbf{x}}, \overline{\mathbf{g}}$ and the matrices $\mathbf{P}_s, \mathbf{P}_t$ containing the eigenvectors of the training data. $\mathbf{b}_s$ and $\mathbf{b}_t$ are the parameters of this parametric deformable model. By discarding eigenvectors that correspond to small eigenvalues an approximated model is formed. This is feasible since the discarded eigenvectors contribute to the lowest variance in the training data. The AAM model incorporates another dimensionality reduction on the concatenated model parameters $\mathbf{b} = (\mathbf{W}_s \mathbf{b}_s, \mathbf{b}_t)^T$ with $\mathbf{W}_s$ being a weighting matrix to relate shape and texture representations. The combined appearance model $\mathbf{b} \approx \mathbf{P}_a \mathbf{a}$ is finally used to generate synthetic model instances by adjusting the appearance parameters $\mathbf{a}$.

AAM model fitting makes use of a learned regression model that describes the relationship between parameter updates and texture residual images. Optimization takes place in a gradient descent scheme using the L_2 norm of the intensity differences (between the synthetic model instances and the given test image) as its cost function and the learned regression model for efficiently approximating the Jacobian of the cost function. The parameters of the cost function are the unknown global pose parameters and the combined appearance parameters (which implicitly define shape and texture parameters). A local minimum of the cost function corresponds to a model fitting solution. Since the minimum is local and the parameter space is very large, multi-resolution techniques have to be incorporated and the fitting requires coarse initialization. Despite the multi-resolution approach, the need

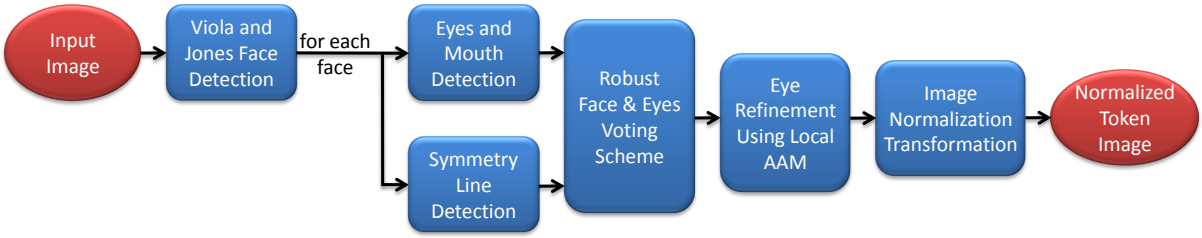


Figure 2: Illustration of the face image normalization work flow consisting of several stages. At the heart of the system is the voting scheme, that decides from a number of face and facial component detections the most probable one while taking the detected symmetry line into account.

for proper initialization marks one of the major drawbacks of this method, with the other drawback being its high sensitivity to image occlusions.

3 Face Image Normalization System Description

In this section we describe our novel algorithm for robust face image normalization. We consider arbitrary input images containing at least one face. However, there are some assumptions we have to make considering potential input images. We restrict the possible deviations in pose angles of the depicted head to smaller than 45 degrees in yaw and pitch. For heads with larger angles the face and facial component detection performance significantly deteriorates, however, in our targeted application of normalizing face images according to the ICAO specification we can safely assume that profile or near-profile images are rare. Roll angle deviations in head pose are targeted in our system by the facial component detection and the face symmetry lines. Another assumption we have to make is that images are taken under normal lighting conditions. Illumination problems that lead to nonlinear intensity changes are therefore not considered.

Our algorithm consists of several modules including face detection, facial component detection, face symmetry line detection, incorporation of spatial a priori knowledge in a probabilistic voting scheme for most likely face and eye detections, a detailed eye center localization module using a statistical model describing shape and appearance, and finally a normalization transformation. The algorithm work-flow is illustrated in Figure 2.

3.1 Face Detection

The face detection component uses the efficient face detector [21] described in Section 2.1. We extend the original feature set according to Lienhart et al. [11] using an open-source implementation [8] that provides a pre-trained detector cascade for portrait and near-portrait faces. After experimentation with this pre-trained face detection cascade we were able to find a parameter set leading to a high detection rate, however, also leaving a significant number of false positive detections. Thus, to achieve an accurate face detection we additionally need to detect facial components to vote for the single face candidate having the highest probability of being the correct face. To prepare for the subsequent probabilistic voting scheme we calculate a normalized confidence measure

$$f = 1 - \exp\left(\frac{\sum_m \alpha_m h_m(\mathbf{x})}{\max_m(\alpha_m)}\right).$$

This measure resembles the confidence of a certain spatial location to be a face.

3.2 Eyes and Mouth Component Detection

We trained facial component detectors using the same AdaBoost object detection (see Section 2.1) framework that was used for face detections. However, the facial components were trained by using our own set of positive and negative object patches. We used 5646 manually annotated face images which were taken from several databases; the Caltech Faces database [1], the FERET database [17], and our own proprietary database. Manual annotations were performed for left and right eye patches and for mouth patches under different poses and different facial expressions (eyes closed, half-closed and open, mouthes closed, open, wide open or smiling). Roughly 40 percent of these images show a deviation from frontal pose and neutral expression. 16 percent of the images contain glasses and 21 percent contain a beard.

For facial component detection we apply these individual detectors to each potential face detection and look for the eyes in the upper two thirds of the face detection, while the mouth search area is restricted to the lower two thirds. This way we receive a number of potential eye and mouth candidates for each face. Again we tune the detection parameters to get a high detection rate with the downside of a larger number of false positive detections, an issue that gets resolved later in our voting scheme. To calculate a normalized confidence measure we calculate the same confidence as described in Section 3.1.

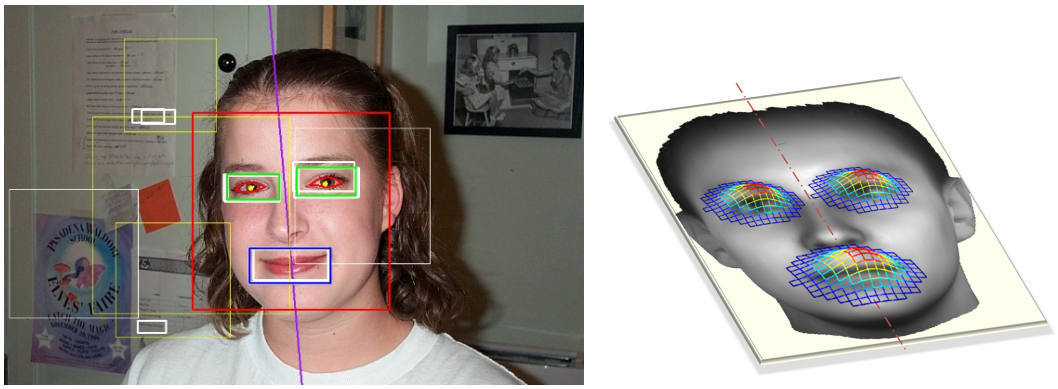
3.3 Face Symmetry Line Detection

Facial mirror symmetry is an important cue that should contribute to a robust face detection system to deal with in-plane rotations. After investigating the state of the art in symmetry detection algorithms we developed a simple and efficient face symmetry detection scheme based on [18]. In this algorithm we calculate image gradient points restricted to the detected face patch. Afterwards we randomly draw a large number of image gradient point pairs, where each of these point pairs votes for a certain symmetry line normal to their connecting line segment. The voting is performed similar to a Hough line voting scheme, taking the similarity of the gradient magnitudes of the point pair into account as the voting increment. Finally the line with the highest vote counter is selected as the potential symmetry line and the magnitude of the vote counts is used as a confidence value. Since this algorithm is rather time-consuming when implemented testing all point pairs, we restrict our search range to point pairs leading to symmetry lines between minus and plus 25 degrees.

3.4 Robust Probabilistic Face and Eye Voting

The previous detection stages lead to a number of face and facial component hypotheses. We proceed with a probabilistic voting framework to find the most likely candidates. Our goal is to effectively combine the different components to form a robust vote for a face detection and an associated eye localization under the constraints of a priori knowledge about spatial locations.

We denote the K potential face rectangle candidates in an image as $F_i, i = 1, \ldots, K$. For each of the K faces, we have candidate rectangles for left eye $L^i_{j_l}, j_l = 1, \ldots, N_l$, right eye $R^i_{j_r}, j_r = 1, \ldots, N_r$ and mouth $M^i_{j_m}, j_m = 1, \ldots, N_m$. Since we have no a priori knowledge about the spatial location of faces in the arbitrary input images, we assume it uniformly distributed. The confidence in a certain face rectangle candidate now depends on two factors. A measure $c_f(F_i)$ delivered from the AdaBoost detection and the contribution of the facial components that were detected inside the face rectangle. For the individual facial component detections per face rectangle we perform a Bayesian probabil-



(a) (b)

Figure 3: Probabilistic face and eye voting: (a) Setup of the voting procedure showing potential face rectangles (white and yellow, white ones have no support by detected facial components) and most likely face (red) due to support of facial components (green, cyan), and (b) prior probability distribution of spatial facial component locations in a face rectangle, modeled as normal distributions with means and covariance matrices derived from manually annotated training data. Image from Caltech Faces database [1].

ity estimation to reach a maximum a posteriori (MAP) solution. Therefore, a priori assumptions $p(L), p(R), p(M)$ on the spatial distribution of eye and mouth patches relative to face rectangles (in a normalized coordinate system to be independent of the size of rectangles) are estimated in a maximum likelihood sense from our database of annotated portrait and near-portrait face images. Thus, the models are normal distributions calculated from the facial component locations in the face coordinate system (see Figure 3b for an illustration). Here the detected symmetry line (compare Section 3.3) is taken into account in order to perform an in-plane rotation of the prior distributions. Furthermore, each facial component detection has an associated likelihood measure from the AdaBoost detection, e.g. for the left eye $p(L_{j_l}|L)$ resembling the likelihood of measurement L_{j_l} being a left eye. To sort out the most likely facial components we apply Bayes theorem to get an MAP solution by comparing the products of priors and likelihoods. For each face F_i we thus receive a left eye rectangle $L^i_{j_l}$ with

$$\arg\max_{L^i_{j_l}} \left(p(L|L^i_{j_l})\right) = \arg\max_{L^i_{j_l}} \left(p(L^i_{j_l}|L)p(L)\right),$$

and right eye $R^i_{j_r}$ and mouth $M^i_{j_m}$ rectangles, respectively. In case no candidate can be found for a facial component from the AdaBoost detection, we hypothesize a candidate at the location of the maximum of the prior distribution. However, this hypothetic candidate gets assigned only a small likelihood of 0.1 since we do not want the guessed part to have a high confidence. If in a face candidate rectangle no facial component can be found at all, we remove the face candidate from further processing. The posteriors for the three facial components now get combined to a single confidence vote by averaging the (normalized) posteriors

$$c_{fc}(F_i) = \frac{1}{3}(p(L|L^i_{j_l}) + p(R|R^i_{j_r}) + p(M|M^i_{j_m})).$$

If no facial component can be found in a face candidate rectangle, we set $c_{fc}(F_i) = 0$. We finally combine the confidences of face rectangle and facial component confidence votes to a single vote using $c(F_i) = c_f(F_i)c_{fc}(F_i)$. The face rectangle F_i maximizing $c\,(F_i)$ is taken as the final result of the face voting, with the corresponding left and right eyes $L^i_{j_l}, R^i_{j_r}$ as final results for the eye localization. This simple scheme gives us robustness to occlusions, since it is not necessary to detect all facial components as long as the final face confidence measure of the most likely face is larger than 0. Figure 3a illustrates this voting scheme.

3.5 Eye Localization Refinement

Our voting procedure results in a robust but rough localization of the eyes in the form of eye patches. In order to come up with an exact localization that can be used to apply the face image normalization procedure, we have to further refine this rough localization. Therefore we have trained an Active Appearance Model as described in Section 2.2. Since we are solely interested in the localization of the eyes, we restrict our model to incorporate only the eyes region (see Figure 4a,b for the setup of the eyes region).

The training data set consists of 427 manually annotated face images taken partly from the Caltech Faces database and partly from our own collection. Training images show variations with respect to open and closed eyes, eye gaze and only slight variations in head pose. During principal component analysis we keep 95 percent of the eigenvalue energy spectrum to represent our compact model on the highest resolution level. We use three levels of resolution and adapt the percentage of kept eigenvectors, i.e., on lower levels we restrict the shape variability to 90 and 80 percent respectively, while keeping the texture and appearance variability at 95 percent. This restriction enables a stronger focus on the global pose parameters for lower resolution levels.

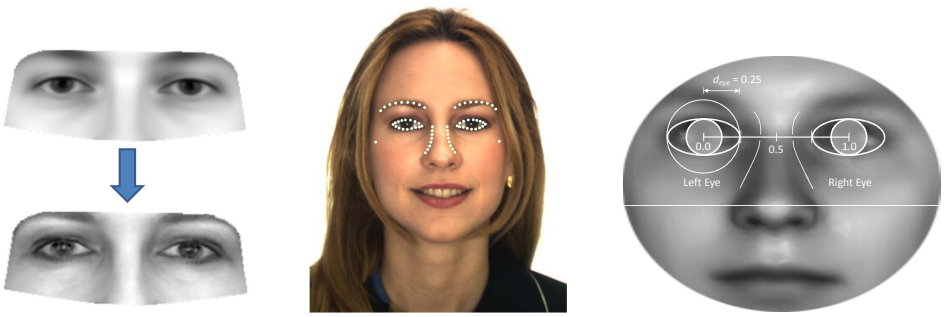
AAM fitting is performed by initializing the mean shape of the AAM with the roughly estimated left and right eye locations from the facial component detection. This gives an initial solution for the pose parameters and the multi-resolution fitting algorithm is started with the lowest resolution. To switch between resolution levels it is necessary to upscale the shape and upsample the texture result and project it into the model of the higher resolution level. This gives the initialization for the higher level. After the fitting procedure has reached a local minimum we report the corresponding eyes shape. Additionally a confidence value derived from the final L_2 norm of the difference between synthetic model texture and warped image texture is calculated.

3.6 Face Normalization Transformation

The final step of our normalization work-flow is the calculation of the face normalization transformation and the resampling of the input image using this transformation according to the ICAO specification of eye locations for normalized token images [9]. Therefore we investigate the confidence value from the AAM model fitting and compare it to a fixed threshold that determines a measure of quality for the fitting. This fixed threshold was determined empirically on a set of typical test images. If the confidence value is large enough we take the center point between the AAM eye corners for left and right eye, respectively, and compute the similarity transformation to map the eye coordinates to the standardized coordinates (see Figure 1). If the confidence in the AAM fitting is too low, we use the center points of the Boosting based eye localization patches as rougher estimates of the true eyes. Additionally if no face candidate with facial components was found, we report that no face was detected. Finally, the resampling is performed using bilinear interpolation. Note that for the evaluation it is only necessary to validate eye locations since the normalization is always the same similarity transformation.

4 Experimental Results

The accuracy of our facial image normalization procedure is tested on three different publicly available databases with known ground truth annotations of the eye centers. Since our intended application is face normalization we focus our evaluation on the eye centers. For evaluation we use the AR face



(a) (b) (c)

Figure 4: Detail illustration of the eye region. (a) Active Appearance Model: Learned mean shape/texture and the texture after successful fitting. (b) AAM shape model after successful fitting drawn on the input image. (c) Eye setup for the experimental evaluation measure *face detection rate*.

database [12] consisting of 509 images, the BioID database [10] consisting of 1521 images, and the IMM database [14] consisting of 240 images. The databases show a variety of different people, poses, occlusions and facial expressions, and they have in common that there exists an annotation of the face intended for use in Active Shape Models where one can easily extract eye center locations from. Our face normalization pipeline leads to a left and right eye center location, we compare this location with the ground truth annotation using the evaluation measure from [10]. We calculate the absolute pixel distance from the ground truth positions to receive two distance values. We choose the larger value and divide by the absolute pixel distance of the two manually set eye positions to become independent from face size according to [10]. We call this value *relative eye distance* d_{eye}. Further, we rate a face as found if the relative distance is equal or less than 0.25, which corresponds to an accuracy of about half the width of an eye in the image. The *face detection rate* is then calculated by dividing the number of correctly found faces by the total number of faces in the dataset (see Figure 4c). Images where no face can be found also contribute to this error measure as mis-detected faces.

We compare our method on all three databases against a standard Viola and Jones face detection approach using the implementation from [8]. For this standard implementation we use the default face detection parameters. We predict the eye location relative to the face rectangle using our large database of manually annotated eye rectangles by combining them with the Viola and Jones face detections. The *relative eye distance* error metric is presented in Figure 5 for the standard Viola and Jones algorithm and our method. Additionally, we compare our results on the BioID database with the face detection results presented by the authors of the BioID study [10] (see Figure 5b), who call their approach *Hausdorff-MLP*. They show quantitative results solely on the BioID database report a *face detection rate* of 91.8% as opposed to 96.1% with our method. On the IMM and AR databases our *face detection rates* are 99.2% and 97.5%, respectively. Concerning the *face detection rate* we see that our method performs significantly better than the Hausdorff distance based method from [10]. This improvement is also reflected in the cumulative error distribution, where the ideal curve would be a step function to 100% at a relative eye distance of zero. The intersection of our proposed method with the relative eye distances of 0.05 and 0.1 are at approximately 62 and 89% compared to 39 and 79% for the *Hausdorff-MLP* method. From the *face detection rate* we can also see that the classic Viola-Jones method performs quite favorable in comparison to our method. However, after looking at the relative error distributions one can clearly see, that the robust method shows a significant improvement in accuracy. This is very important since the definition of the face detection rate [10] allows a large amount of error. We conclude that our method outperforms the state of the art algorithms on these

three publicly available data sets. Finally Figure 6 shows a number of qualitative results for the eye detection in the presence of pose deviations, facial expressions, and occlusions. The rightmost image of the bottom row shows a detection problem due to a pose deviation that is too large. Here also the symmetry detection has failed. The leftmost image of the bottom row shows an occluded eye, which is correctly detected by the probabilistic voting scheme. Images where no refined eye contour is drawn have resulted in a bad AAM fit, such that the center of the AdaBoost eye rectangles are used as eye centers. The other images show correct algorithm results. Finally, to assess algorithm run-time we did an evaluation on a small set of typical input images. The Viola-Jones implementation requires on average about 900 ms to detect a face from 800×600 images (note that we use the detection cascade on a large number of scales and with conservative parameter settings) on an Intel Core2 Duo notebook CPU at 2.53 GHz. Our algorithm performs of course slower due to the robust extensions. However, our run-time of on average about 3300 ms is still acceptable for our task of face image normalization.

5 Conclusion

This work presents a robust face normalization system suitable for transforming arbitrary images containing a face into a standardized coordinate system as e.g. needed to fulfill the ICAO requirements on passport photographs. We show how our robust system is able to process input images with pose deviations, occlusions and over different facial expressions by making use of several redundant component detection methods. A robust approach is achieved by combining these detections. Our experimental results show superior results compared to state of the art algorithms. Further work is necessary to increase the range of pose deviations we are able to process. Here we need multi-view face and facial component detections which will be incorporated using multiple separately trained detection cascades over the possible range of poses.

Acknowledgement

This work has been funded by the Biometrics Center of Siemens IT Solutions and Services, Siemens Austria.

References

[1] Caltech. Caltech Faces database. http://www.vision.caltech.edu/html-files/archive.html, 1999.

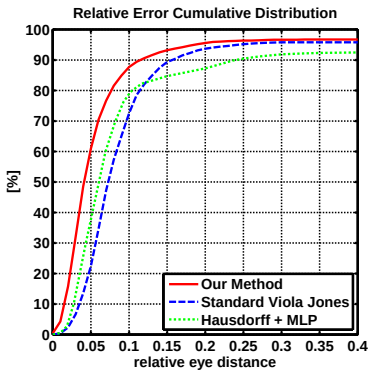
[2] T. F. Cootes, G. J. Edwards, and C. J. Taylor. Active Appearance Models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(6):681–685, June 2001.

[3] V. Erukhimov and K.-C. Lee. A bottom-up framework for robust facial feature detection. In *Proc. 8th IEEE International Conference on Automatic Face and Gesture Recognition*, 2008.

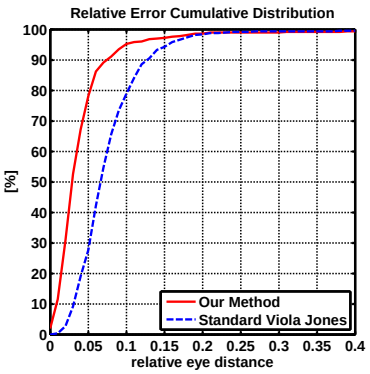
[4] P. F. Felzenszwalb and D. Huttenlocher. Pictorial structures for object recognition. *International Journal of Computer Vision*, 61(1):55–79, 2005.

[5] Y. Freund and R. E. Schapire. A short introduction to boosting. *Journal of Japanese Society for Artificial Intelligence*, 14(5):771–780, September 1999.

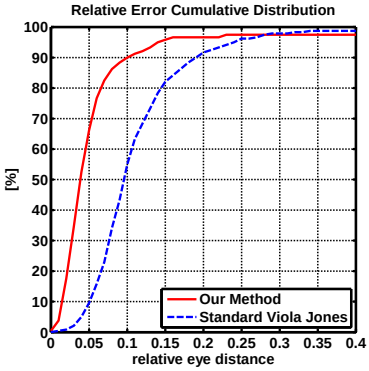
[6] B. Heisele, T. Serre, M. Pontil, and T. Poggio. Component-based face detection. In *Proc. IEEE Int. Conf. Computer Vision and Pattern Recognition*, pages 657–662, 2001.



(a)



(b)



(c)

Figure 5: Quantitative results on relative error distributions of our method compared to different algorithms. (a) BioID database (1521 images), (b) AR database (509 images), (c) IMM database (240 images).

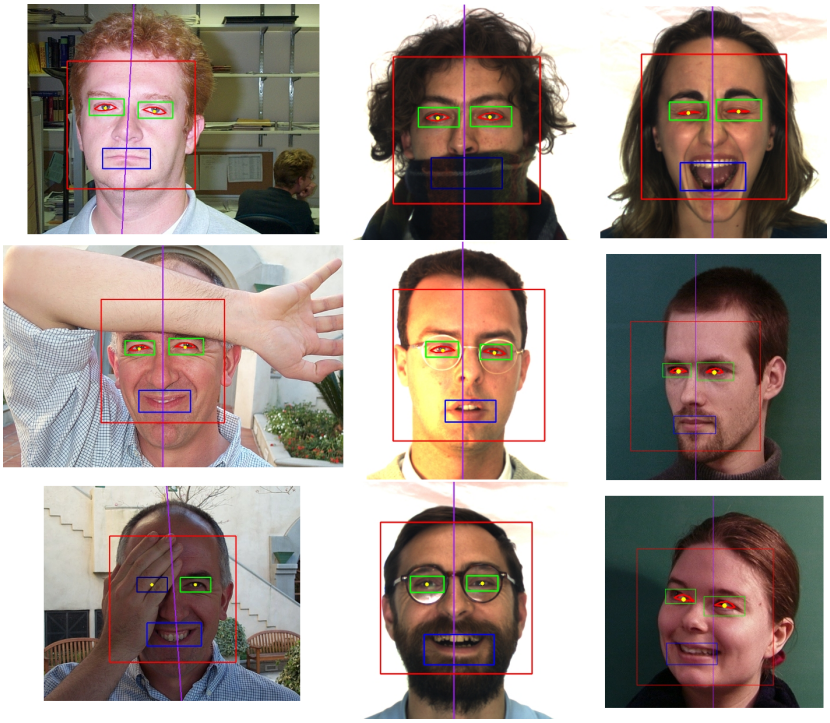


Figure 6: Selected qualitative results on representative images from the databases.

[7] E. Hjelmas and B. K. Low. Face detection: A survey. *Computer Vision and Image Understanding*, 83(3):236–274, Sep 2001.

[8] Intel. Open computer vision (OpenCV) library. http://sourceforge.net/projects/opencv, 2007.

[9] ISO International Standard ISO/IEC JTC 1/SC37 N506. Biometric data interchange formats - part 5: Face image data, 2004.

[10] O. Jesorski, K. Kirchberg, and R. Frischholz. Robust face detection using the hausdorff distance. In J. Bigun and F. Smeraldi, editors, *Audio and Video based Person Authentication - AVBPA 2001*, pages 90–95, Halmstad, Sweden, 2001. Springer.

[11] R. Lienhart and J. Maydt. An extended set of haar-like features for rapid object detection. In *Proc Int Conference on Image Processing (ICIP)*, volume I, pages 900–903, 2002.

[12] A. Martinez and R. Benavente. The AR face database. Technical Report 24, CVC, June 1998.

[13] B. Moghaddam and A. Pentland. Probabilistic visual learning for object representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(7):696–710, July 1997.

[14] M. M. Nordstrøm, M. Larsen, J. Sierakowski, and M. B. Stegmann. The IMM face database - an annotated dataset of 240 face images. Technical report, Informatics and Mathematical Modelling, Technical University of Denmark, DTU, 2004.

[15] M. Pantic and L. Rothkrantz. Automatic analysis of facial expressions: The state of the art. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(12):1424–1445, 2000.

[16] P. J. Phillips, W. T. Scruggs, A. J. O'Toole, P. J. Flynn, K. W. Bowyer, C. L. Schott, and M. Sharpe. FRVT 2006 and ICE 2006 large-scale results. Technical report, National Institute of Standards and Technology, Gaithersburg, MD 20899, March 2007.

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

[18] V. S. N. Prasad and B. Yegnanarayana. Finding axes of symmetry from potential fields. *IEEE Transactions on Image Processing*, 13(12):1559–1566, Dec 2004.

[19] H. A. Rowley, S. Baluja, and T. Kanade. Neural network-based face detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(1):23–38, Jan. 1998.

[20] H. Schneiderman and T. Kanade. A statistical method for 3d object detection applied to faces and cars. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, volume 1, pages 746–751, 13–15 June 2000.

[21] P. Viola and M. J. Jones. Robust Real-Time Face Detection. *International Journal of Computer Vision*, 57(2):137–154, 2004.

[22] M.-H. Yang, D. J. Kriegman, and N. Ahuja. Detecting faces in images: a survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(1):34–58, Jan. 2002.

[23] W. Zhao, R. Chellappa, P. J. Phillips, and A. Rosenfeld. Face recognition: A literature survey. *ACM Computing Surveys*, 35(4):399–458, Dec 2003.