3

Automatic Facial Feature Extraction for Face Recognition

Paola Campadelli, Raffaella Lanzarotti and Giuseppe Lipori
Università degli Studi di Milano
Italy

1. Introduction

Facial feature extraction consists in localizing the most characteristic face components (eyes, nose, mouth, etc.) within images that depict human faces. This step is essential for the initialization of many face processing techniques like face tracking, facial expression recognition or face recognition. Among these, face recognition is a lively research area where it has been made a great effort in the last years to design and compare different techniques.

In this chapter we intend to present an automatic method for facial feature extraction that we use for the initialization of our face recognition technique. In our notion, to extract the facial components equals to locate certain characteristic points, e.g. the center and the corners of the eyes, the nose tip, etc. Particular emphasis will be given to the localization of the most representative facial features, namely the eyes, and the locations of the other features will be derived from them.

An important aspect of any localization algorithm is its precision. The face recognition techniques (FRTs) presented in literature only occasionally face the issue and rarely state the assumptions they make on their initialization; many simply skip the feature extraction step, and assume perfect localization by relying upon manual annotations of the facial feature positions.

However, it has been demonstrated that face recognition heavily suffers from an imprecise localization of the face components.

This is the reason why it is fundamental to achieve an automatic, robust and precise extraction of the desired features prior to any further processing. In this respect, we investigate the behavior of two FRTs when initialized on the real output of the extraction method.

2. General framework

A general statement of the automatic face recognition problem can be formulated as follows: given a stored database of face representations, one has to identify subjects represented in input probes. This definition can then be specialized to describe either the *identification* or the *verification* problem. The former requires as input a face image, and the system determines the subject identity on the basis of the database of known individuals; in the latter situation the system has to confirm or reject the identity claimed by the subject.

Source: Face Recognition, Book edited by: Kresimir Delac and Mislav Grgic, ISBN 978-3-902613-03-5, pp.558, I-Tech, Vienna, Austria, June 2007

As noted by [Zhao et al., 2003], whatever the problem formulation, its solution requires the accomplishment of three subsequent subtasks: *face detection*, *feature extraction* and *face recognition* (Figure 1).

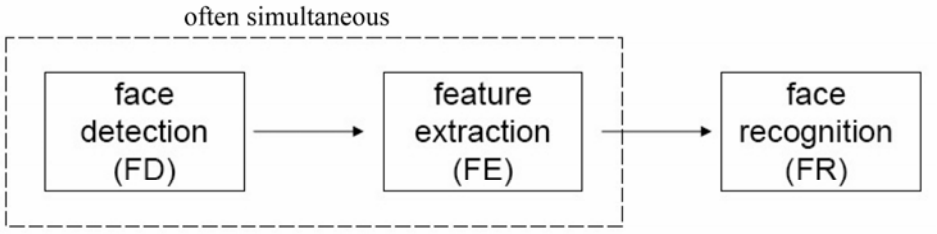

Figure 1. The subtasks of the face recognition problem

In fact, given an input image depicting one or more human subjects, the problem of evaluating their identity boils down to detecting their faces, extracting the relevant information needed for their description, and finally devising a matching algorithm to compare different descriptions.

On one hand, the modularity of the original problem is a beneficial property as it allows to decompose it and to concentrate on the specific difficulties of each task in order to achieve a more effective solution. On the other hand, care must be taken in recomposing the separate modules: a common approach is to devise techniques that face only a task at once[1] without considering the problems that can arise at the "interfaces" between them.

In particular, most of face recognition techniques (FRTs) presented in literature skip the previous tasks and assume perfect feature extraction. While this can be certainly useful to develop and compare different recognition strategies, this attitude is not practical if the goal is to produce a fully automatic recognition system. Relying upon manual annotations of the feature positions does not account for the influence played by the extraction error on the recognition rate: the amount and trend of this dependency is not easily predictable and varies from FRT to FRT.

These facts bring to two important observations: first of all it is fundamental to achieve an automatic, robust and precise extraction of the desired features prior to the application of a face recognition technique; secondly, it is important to study the relation between the quality of the feature extraction and the performance of the face recognition. By doing so, one ensures to couple only truly compatible modules to realize a fully automatic, robust system for face recognition. Differently stated, any FRT should be aware of the minimum precision required for its functioning and should clearly declare it.

Regarding feature extraction, there is a general agreement that eyes are the most important facial features, thus a great research effort has been devoted to their detection and localization [Ji et al., 2005, Zhu and Ji, 2005, Fasel et al., 2005, Hamouz et al., 2005, Tang et al., 2005, Wang et al., 2005, Song et al., 2006, Gizatdinova and Surakka, 2006]. This is due to several reasons, among which:

- eyes are a crucial source of information about the state of human beings.

---

[1] Face detection and feature extraction are often accomplished simultaneously as it is possible to locate faces by directly locating their inner features.

- the eye appearance is less variant to certain typical face changes. For instance they are unaffected by the presence of facial hair (like beard or mustaches), and are little altered by small in-depth rotations and by transparent spectacles.
- the knowledge of the eye positions allows to roughly identify the face scale (the inter-ocular distance is relatively constant from subject to subject) and its in-plane rotation.
- the accurate eye localization permits to identify all the other facial features of interest.

To our knowledge, eyes are the only facial features required for the initialization of *any* FRT; actually this is the only information needed by those methods that operate an *alignment* of the face region, for instance as done by [Zhang et al., 2005]. However some techniques may require more features than just the eyes. For instance all FRTs derived from subspace methods (see [Shakhnarovich and Moghaddam, 2004] for a detailed survey) are initialized on four positions (the eyes, nose and mouth locations) to *warp* the face region before projection.[2] Other techniques operate on larger sets of facial positions because they base the recognition on some kind of local processing: e.g. [Wiskott et al., 1999] is based on the comparison of the image texture found in the neighborhood of several *fiducial points*.
Due to these considerations, the performance evaluation of a feature extraction method is usually given in terms of error measures that take into account only the localized eye positions. In Sec. 3. we will motivate the choice of such measures and we will introduce the study of the recognition rate in function of the eye localization precision. Sec. 4. presents the proposed algorithm for precise eye localization, together with the experimental results of its application on many public databases. In Sec. 5. we show a possible way to automatically derive the locations of a set of facial features from the knowledge of the sole eye positions. Sec. 6. reports the results of two face recognition experiments carried out on automatically extracted features: the behavior of two FRTs is discussed by making some considerations about their dependence on the extraction quality.

3. The importance of precise eye localization

Given the true positions of the eye centers (by manual annotation), the eye localization accuracy is expressed as a statistics of the error distribution made over each eye (usually the mean or the maximum), measured as the Euclidean pixel distance. In order to make these statistics meaningful, so that they can be used to compare the results obtained on any dataset, it is necessary to standardize the error by normalizing it over the face scale.
One popular error measure has been introduced by [Jesorsky et al., 2001], and it has been already adopted by many research works on eye localization. The measure, which can be considered a worst case analysis, is defined as

$$d_{eye} = \frac{\max(||C_l - \tilde{C}_l||, ||C_r - \tilde{C}_r||)}{||C_l - C_r||}$$

[2] Both the alignment and the warping are operations that intend to normalize a face database. The former consists in bringing the principal features (usually the eyes) to the same positions. This is done via an affine transformation (a scaling plus a roto-translation) that uses the eye centers as "pivots" of the transform. A warping is a non-affine transformation (a non uniform "stretching" of the face appearance) that is meant to densely align the face appearance (or at least the position of several features).

where (C_l, C_r) are the ground truth positions and $(\tilde{C}_l, \tilde{C}_r)$ the results of automatic localization. There is a general agreement [Jesorsky et al., 2001, Ma et al., 2004a, Zhou and Geng, 2004] that $d_{eye} \leq 0.25$ is a good criterion to flag the eye presence (to claim eye detection). This precision roughly corresponds to a distance smaller than or equal to the eye width. However, this accuracy level may not be sufficient when the localized positions are used for the initialization of subsequent techniques.

Following the idea presented in [Ma et al., 2004a], we studied the relation between d_{eye} and the face recognition rate of some baseline methods available in the CSU package [Beveridge et al., 2005] together with the LAIV-FRT described in Sec. 6. To mimic the behavior of eye localization techniques that achieve different levels of precision, we carried out four recognition experiments by artificially perturbing the ground truth quality; both C_r and C_l have been randomly displaced inside circles of radii equal to 5%, 10% and 15% of $\|C_l - C_r\|$ with uniform distribution. In Figure 2 we report the results of this study on the XM2VTS database (see Appendix 8.). The experiment is defined as follows: session 1 is used for the gallery, session 2 for the probe, sessions 3 and 4 constitute the training set.[3] Differently from [Ma et al., 2004a] where only the probe set is affected by artificial error, all three sets (gallery, probe and training) have been perturbed as it would happen in a completely automatic system. The graphs of Figure 2 clearly show that the precision of eye localization is critical for the alignment of faces, even if it does not affect all the methods in the same way.

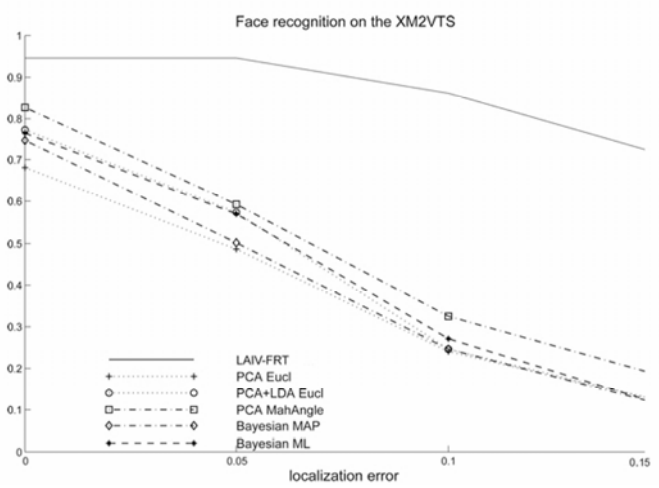

Figure 2. Face recognition vs. (artificial) eye localization precision

Very recently in [Rodriguez et al., 2006] the issue has been further developed, suggesting a new error measure which is more discriminative than d_{eye} as it permits a quantitative evaluation of the face recognition degradation with respect to different error types. Instead of considering only the Euclidean distance between the detections and the ground truth points, it considers four kinds of error: the horizontal and the vertical error (both measured

[3] The training set is needed by all the reported CSU methods, not by LAIV-FRT.

between the mid-points $C_o, \tilde{C}_o$ of the segments $\overline{C_r C_l}$, $\overline{\tilde{C}_r \tilde{C}_l}$, see Figure 3), the scale and the rotation error.

$$\Delta_x = \frac{dx}{||C_l - C_r||} \quad (horizontal) \qquad \Delta_s = \frac{||\tilde{C}_l - \tilde{C}_r||}{||C_l - C_r||} \quad (scale)$$

$$\Delta_y = \frac{dy}{||C_l - C_r||} \quad (vertical) \qquad \Delta_\alpha = \overrightarrow{C_l C_r} \overrightarrow{\tilde{C}_l \tilde{C}_r} \quad (rotation)$$

In fact it happens that some FR systems are more sensitive to certain types of error. In particular, the baseline PCA method is extremely sensitive to all types, while the FR system described in the article (referred to as DCT/GMM) seems to be almost indifferent to translational errors (Δ_x, Δ_y), while its performance notably degrades when the error is due principally to scale or rotation inaccuracy (Δ_s, Δ_α). The authors conclude that it is not possible to define an absolute concept of precise localization: each FR will have a different tolerance to errors and it should clearly state the level and type of precision required for its initialization.

The article [Shan et al., 2004] is entirely devoted to the so called *curse of misalignment*. There it is reported the high dependence of the Fisherface method [Belhumeur et al., 1997] performance on the alignment precision, especially with respect to rotation or scale errors. The authors also propose to evaluate the *overall face recognition rate* with a measure, *rate**, that integrates the FR rate over all possible misaligned initializations, weighted by their probability:

$$rate^* = \int_{e \in \text{errors}} rate(e) P(e) \mathrm{d}e \tag{1}$$

They measure the robustness of a FRT to errors as the overall FR rate normalized with respect to the ideal case of absence of error, i.e. *rate**/*rate*(0). Although we deem correct the definition of the overall FR rate, the limit of this approach is the difficulty of knowing the pdf of the misalignment distribution, thus preventing from a direct computation of *rate**.

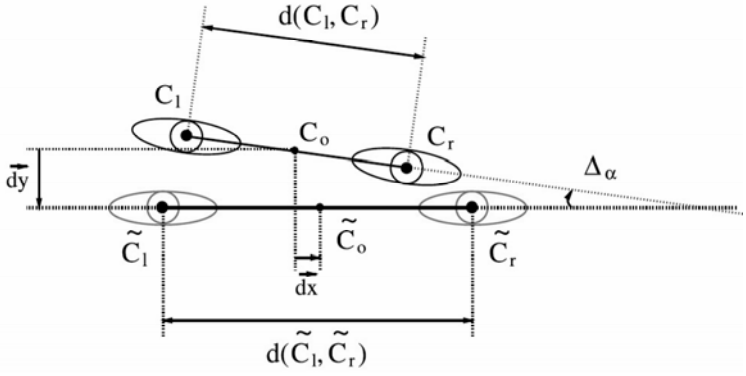

Figure 3. Localization error: (C_l, C_r) are the ground truth positions, $(\tilde{C}_l, \tilde{C}_r)$ are the results of automatic localization

A totally different approach is that of [Martinez, 2002] where, instead of imposing the maximum level of acceptable localization error, it is proposed to deal with it by learning its

distribution directly into the statistical model of each subject. The method requires a quantitative estimate of the localization error distribution to be used to perturb each image accordingly, generating a certain number of new images constituting the set of all the possible displacements. These enriched samples become the classes to be modelled (one for each subject). Such models are then used for face recognition, being robust to localization errors by construction. A similar approach has also been proposed by [Min et al., 2005].

4. Coarse-to-fine eye localization

The general outline of our eye localization system is presented in Figure 4. The system assumes to be initialized on a *face map* (a binary image of the regions that have been detected as faces) and processes it in a coarse-to-fine fashion: the first level is an eye detector meant to locate the eye pattern; the second level is initialized on the positions output by the first one and aims at improving the localization precision. Both modules are based on strong statistical classifiers and both take advantage of a suitable eye representation consisting in optimally selected wavelet coefficients. One important difference lies in the definition of the receptive field of the respective eye patterns: the first is equal to the inter-ocular distance, while the second is half of it to consider a finer space resolution (see some examples in Figure 5).

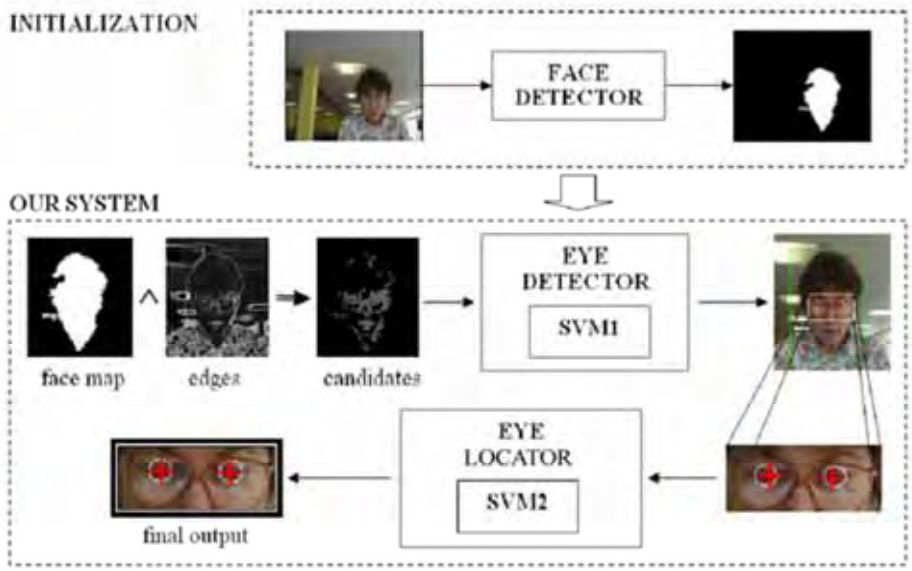

Figure 4. General outline of the eye localization system

The system can be applied to the output of any face detector that returns a rough estimation of the face position and scale, e.g. [Viola and Jones, 2004, Schneiderman and Kanade, 2004, Osadchy et al., 2005, Campadelli et al., 2005]. The eye detector serves two distinct objectives: it not only produces a rough localization of the eye positions, it also validates the output of the face detector (a region of the face map is validated as a true face if and only if there has

been at least an eye detection within it). In fact all face detectors manifest a certain false detection rate that must be dealt with.

Figure 5. Examples of eye patterns for the eye detector (first row) and locator (second row)

4.1 Wavelet selection

The difficulty intrinsic to the task of eye localization requires an accurate choice of a suitable representation of the eye pattern. It has been observed that the wavelet representation is more favorable than the direct representation as it leads to a smaller generalization error [Huang and Wechsler, 1999]. Haar-like wavelets permit to describe visual patterns in terms of luminance changes at different frequencies, at different positions and along different orientations.

Before the wavelet decomposition, each eye patch undergoes an illumination normalization process (a contrast stretching operation) and is then reduced to 16×16 pixels.[4] The decomposition is realized via an *overcomplete* bi-dimensional FWT (Fast Wavelet Transform) [Campadelli et al., 2006a] that produces almost four times as many coefficients with respect to the standard FWT. This redundancy is desirable as we want to increase the cardinality of the feature "vocabulary" before going through the selection procedure.

In order to carry out the feature selection, we follow the idea proposed in [Oren et al., 1997] to apply a normalization step, which allows us to distinguish two sub-categories of wavelet coefficients: C^+ and C^-. Both retain precious information: the first class gathers the coefficients that capture the edge structure of the pattern, while the second class contains the coefficients that indicate a systematic absence of edges (in a certain position, at a certain frequency and along a certain orientation). What is more important, the normalization step naturally defines a way to (separately) order the two categories, thus providing a way to assess the relative importance of the respective coefficients (for the technical details refer to [Campadelli et al., 2006b]).

Once ordered the normalized coefficients, we define an error function to drive the selection process. We can measure the expressiveness of the coefficients by measuring how well they reconstruct the pattern they represent. We wish to find the set of optimal coefficients

$$w = \arg \min_{\substack{w = w^+ \cup w^-, \\ w^+ \subseteq C^+,\ w^- \subseteq C^-}} ||E - E_w||^2 + \alpha \cdot ||E_w - U||^2 \tag{2}$$

[4] Such a dimension represents a trade off between the necessity to maintain low the computational cost and to have sufficient details to learn the pattern appearance.

where E is the mean eye pattern.[5] U is the uniform pattern (with all pixels set to the mean luminance of E) and E_w is the reconstruction obtained by retaining the set w of the wavelet coefficients $w^+ \subseteq C^+$ and $w^- \subseteq C^-$. The first term of the objective function represents the error made by the reconstruction, while the second term intends to bound the amount of detail we are adding to the pattern representation (the value α is a trade-off to balance between these two opposite goals). The ordering of the coefficients avoids to optimize over all the possible subsets of $C^+ \cup C^-$: w is incremented by iteratively adding new coefficients according to their ordering.

We experimentally observed that the trend of the objective function is rather insensitive to variations of α in the interval [0.5, 1]; we set it to 0.8. As it can be expected, the norm of the reconstruction maximally varies increasing the number of w^+ retained, while it is almost unaffected by the number of selected w^-. Due to this consideration, the selected $w = w^+ \cup w^-$ are such that they correspond to a local minimum of the objective function (2.), with the additional constraint $|w^+|/|C^+| \sim |w^-|/|C^-|$.

Figure 6 shows the coefficients selected for the pattern representation of each classifier. For the eye detector the process retains 95 wavelet coefficients that well characterize the general eye shape (the highest frequency coefficients are not considered). The representation associated with the eye locator keeps 334 coefficients, therefore the application of the second classifier is more costly than the first one.

Figure 6. From left to right: the mean eye pattern, its wavelet decomposition and the selected features (red contour) of the two eye patterns. High intensities correspond to strong edges, low intensities indicate uniform regions

4.2 Eye detection

The module for eye detection takes in a face map output by a generic face detector and produces a first, rough localization of the eye centers. Its core component is a strong statistical classifier that is capable of distinguishing the eye appearance from that of the other facial features; for this purpose we employ a binary Support Vector Machine (SVM), that is the state-of-the-art model for many classification tasks [Vapnik, 1995]. The classification is carried out on examples represented via a set of 95 selected wavelet filter responses, as described in the previous section.

The training of the SVM has been carried out on a total of 13591 examples extracted from 1416 images: 600 belonging to the FERET database (controlled images of frontal faces), 416 to the BANCA database (to model different illumination conditions and the closed eyes), and 600 taken from a custom database containing many heterogenous and uncontrolled

[5] Defined simply by averaging the gray levels of 2152 eye patterns.

pictures of various people (useful to model pose variations, non-neutral face expressions and random background examples). The positive class is built to contain eye examples cropped to a square of side equal to the inter-ocular distance. The negative class is populated by the other facial features (nose, mouth, chin, cheeks, forehead, etc.) and by some examples extracted from the background of images (respectively 3 and 2 for every positive). The definition of the two classes is driven by the notion that the eye detection module must be applied most of the time within the face region, therefore a negative example in this context is actually a facial feature distinct from the eyes. However, as face detectors sometimes detect some false positives, it is useful to enrich the definition of the negative class by adding random negative patterns.

The machine is defined as follows: we employed a *C*-SVM (regulated by the error-penalization parameter *C*) based on the RBF kernel (parameterized by $\gamma = \frac{1}{2\sigma^2}$, which regulates the amplitude of the radial supports). The tuning of the two hyper-parameters *C* and γ has been done in order to maximize the *precision* × *recall*[6] on a test set of 6969 examples disjoint from the training set, but generated according to the same distribution. This procedure selected $C = 6$ and $\gamma = 4.0 \times 10^{-4}$, which yielded a SVM of 1698 support vectors (let us call it SVM1) and a 3.0% of misclassifications on the test set. This error can be considered an empirical estimate of the generalization error of the binary classifier.

Once trained, the SVM1 is integrated into a pattern search strategy that avoids a multiscale scan: we infer the size of a hypothetical eye present in that region from the size of the face detector output.[7] However, any face detector is subject to a certain error distribution on the size of its detections (either over-estimating or under-estimating the true face size), so the inferred eye scale cannot be fully trusted. We account for this uncertainty by considering a range of three scales; the evaluation of a candidate point *P* comes down to evaluating three examples centered in it: the one at the inferred scale ($\mathbf{x}_P$), plus two examples ($\mathbf{x}_{P^-}$ and $\mathbf{x}_{P^+}$) extracted in a way to account for an error distribution of the face size that is between half and twice the true size. This is a very reasonable requirement for a good face detector and permits to treat almost all of its outputs. If SVM1($\mathbf{x}$) = 0 is the equation of the decision function (hyperplane) separating the two classes, then we can treat the functional margin SVM1($\mathbf{x}$) as a "measure" of the confidence with which the SVM classifies the example $\mathbf{x}$. Thus we define the function

$$\rho(P) = SVM1(\mathbf{x}_P) + SVM1(\mathbf{x}_P^-) + SVM1(\mathbf{x}_P^+)$$

as the strength of the candidate point *P*.

Moreover, in order to make the search more efficient, we avoid an exhaustive scan of the candidate points: first comes the identification of points lying on edges, then they are subsampled with a step that depends on the scale of the face region;[8] we consider as detections the points for which $\rho(P) > 0$, and we group them according to their proximity in

---

[6] If TP = true positives, FN = false negatives, FP = false positives

$$precision = \frac{TP}{TP + FP} \qquad recall = \frac{TP}{TP + FN}$$

[7] This relation has been estimated for each employed face detector and applied consistently.

[8] The subsampling step is defined as $\lceil \frac{\text{region radius}}{25} \rceil$, where the "radius" of a region is simply $\sqrt{\frac{\text{area}}{\pi}}$.

the image;[9] each group of point candidates is then represented by its centroid (the eye center) obtained weighting each point P with its $\rho(P)$.
Ideally we should have just two eye centers detected for each face, however sometimes it happens that the eye classifier detects also one or more false positives. To deal with this, we introduce a selection criterion that exploits the margin of the classifier and assumes the substantial verticality of the face pose. Doing so, we manage to select the eye positions, and to discard the false detections, by choosing the couple of centers (c_i, c_j) that maximizes

$$\frac{SVM(c_i) \cdot SVM(c_j)}{1 + \sqrt{|(c_i)_y - (c_j)_y|}}$$

where $(c_i)_y$ is the y coordinate of the center c_i. As we do not want to enforce the perfect verticality of the face, the square root at denominator is introduced to give more importance to the strength of the eye centers with respect to their horizontal alignment.
Figure 7 visualizes the data flow of the eye detection module.

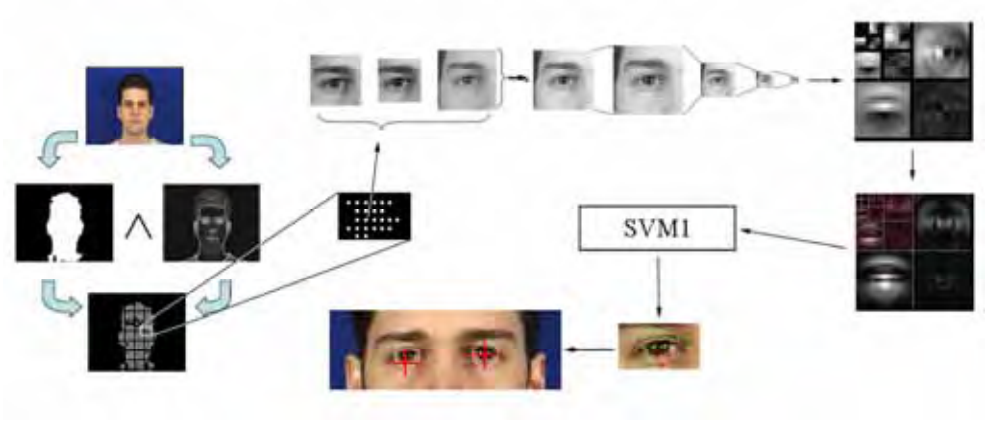

Figure 7. Eye detector outline

4.3 Eye localization

The module for eye localization is conceived to be applied in cascade to the eye detection one, when it is desirable a greater localization precision of the detected positions. The general architecture of this module is very similar to the previous one, therefore we can concentrate on the description of the main differences.
While the eye detector must distinguish the global eye shape from that of other facial patterns, the eye locator must work at a much finer detail level: the goal here is to start from a rough localization and refine it by bringing it closer to the exact eye center location. Bearing in mind this objective, at this stage we consider a richer pattern representation (334 wavelet coefficients) that permits a finer spacing resolution. The positive examples

[9] Two detections are "close", and hence must be aggregated, if their Euclidean distance is smaller than five times the subsampling step. This multiple is not arbitrary, as it corresponds to about half the distance between the eye corners.

correspond to a smaller receptive field (half of the inter-ocular distance) and the negative examples are generated by small, random displacements of the subimages used for the extraction of the positive ones (10 negative examples for each positive).
The *C*-SVM with RBF kernel is first tuned in the same way as before, selecting $C = 1.35$ and $\gamma = 3.6 \times 10^{-4}$. The training is then carried on over 22647 examples, producing a SVM of 3209 support vectors (SVM2 from now on) that exhibits a misclassification rate of 2.5% on a test set of 11487 examples.
The output of the eye detection module is used for the initialization of the eye localization module. The pattern search proceeds only in a small neighborhood of the starting locations, but this time we do an exhaustive scan as we do not want to loose spacial resolution. The search is done at only one scale, inferred averaging the three scales previously considered and weighting them according to their respective SVM1 margin (the factor $\frac{1}{2}$ is due to the smaller receptive field):

$$\frac{1}{2} \times \frac{\sum_{\mathbf{x} \in \{\mathbf{x}_P, \mathbf{x}_P^+, \mathbf{x}_P^-\}} [\Theta(SVM1(\mathbf{x})) \times (\text{scale of } \mathbf{x})]}{3} \text{ where } \Theta(z) = \begin{cases} z & \text{if } z > 0 \\ 0 & \text{if } z \leq 0 \end{cases}$$

Finally the SVM2 evaluations are thresholded at 0, determining a binary map consisting of one or more connected regions. The refined eye center is found at the centroid of the connected region that weights the most according to the SVM2 margin.
Figure 8 visualizes the data flow of the eye localization module.

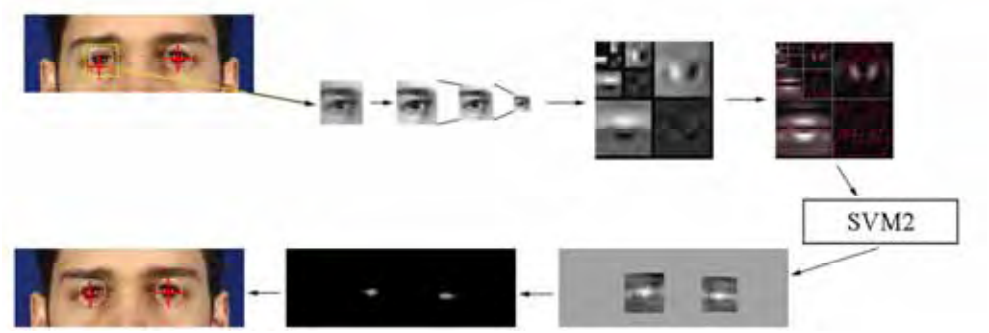

Figure 8. Eye locator outline

We note here that the computational cost of each single SVM evaluation is linearly proportional to the number of support vectors. Therefore, in order to reduce the computational time of our application, it would be desirable to approximate the hyperplane associated to the SVM by reducing the number of its supports, without deteriorating its separation abilities. Some research has been devoted to optimal approximation techniques for support vector reduction, which usually require to specify aforetime the desired number of supports to retain at the end of the reduction process [Burges, 1996, Schölkopf et al., 1999]. However there is no general rule regarding how many vectors can be suppressed before compromising the performance of a SVM classifier; this quantity clearly depends on the difficulty of the classification task. Another approach consists in fixing a threshold on the maximum marginal difference of the old support vectors with respect to the new hyperplane [Nguyen and Ho, 2005]. This perspective is particularly interesting as it enables

to specify a stop quantity that is no more arbitrary, on the contrary it allows to limit the oscillation of the decision surface.
We have reimplemented the technique described in [Nguyen and Ho, 2005] and applied it only to the SVM2 because a reduction of this machine would be of great benefit with regards to the computational time: in fact it is composed of almost twice as many support vectors than the SVM1, and it is evaluated at many more candidate points. What is more, while a reduction of the SVM1 strongly influences the eye detection rate, a reduced SVM2 only degrades the localization precision, and in a much more progressive way. The results of the reduction experiments are given in the next section.

4.4 Eye localization results

The experiments have been carried out on images taken from the following datasets: XM2VTS, BANCA, FRGC v.1.0, BioID and FERET (see Appendix 8. for the full specification of the datasets composition). All these images depict one subject shot with vertical, frontal pose, eyes closed or open, presence or absence of spectacles; none of these images has been used for the training of the SVM classifiers. On color images (XM2VTS, BANCA, FRGC) the face detection has been carried out using the method in [Campadelli et al., 2005], while when the input images are gray scale (BioID, FERET), the detection is performed by a re-implementation of [Viola and Jones, 2001].

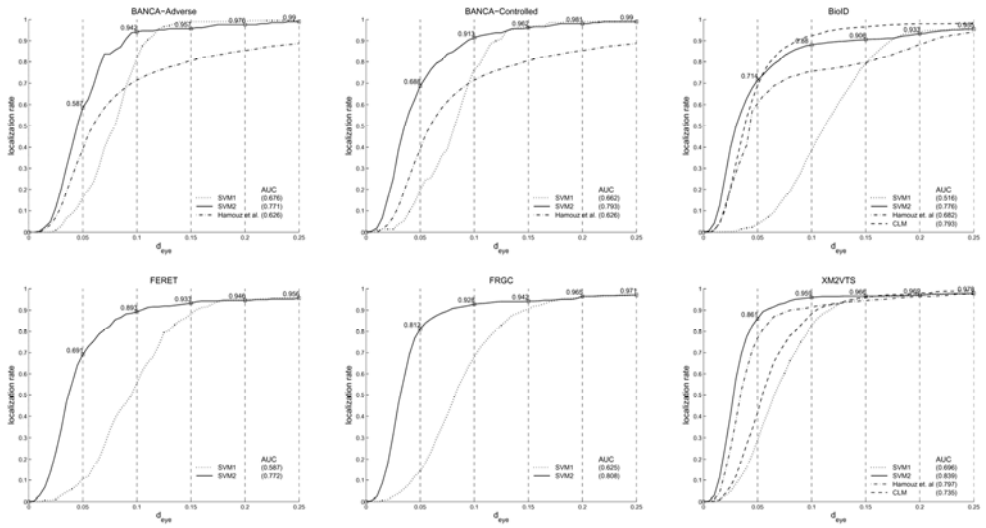

Figure 9. The cumulative distributions of eye detection and localization over different databases

The graphs in Figure 9 display the performance of the eye detector (SVM1), the eye locator (SVM2) and, when available, we report the performance achieved by the methods presented by [Hamouz et al., 2005] (denoted as "1 face on the output" in the original article) and [Cristinacce and Cootes, 2006] (Constrained Local Models, CLM). Regarding CLM, the curves plotted on the BioID and XM2VTS graphs have been extrapolated from the results kindly provided by the authors of the method.

The numbers reported in parenthesis on the graphs represent the Area Under the Curve (AUC), therefore they give a global estimation of the performance of each localization method over that particular dataset. Regarding eye detection, the SVM1 alone permits to achieve rates of 99.0%, 95.5%, 95.6%, 97.1% and 97.8% over the datasets BANCA, BioID, FERET, FRGC and XM2VTS respectively ($d_{eye} \leq 0.25$). As expected, the addition of the second classifier greatly improves the precision of the detection and the curves are systematically above the rates declared by Hamouz et al. Regarding CLM, we note that it is very effective in localizing the eyes over the BioID database, while on the XM2VTS it achieves a lower rate.[10]

Also the works by [Jesorsky et al., 2001], [Ma et al., 2004b], [Tang et al., 2005] and [Niu et al., 2006] use the error measure d_{eye} in order to assess the quality of eye localization. The first work exhibits a localization performance that is lower than that reported by Hamouz et al. The second one presents a cumulative curve that looks similar to the performance of the SVM1 but it is obtained referring to a mix of databases with no intersection with the ones we considered, making impossible a direct comparison. The third paper reports results on the BioID, tabulating only the values corresponding to $d_{eye} \leq 0.1$ and $d_{eye} \leq 0.25$ (91.8% and 98.1% respectively), while omitting the curve behavior under this value. Finally, the last work presents results on XM2VTS and BioID; we do not report them in figure since the values are not clearly tabulated, however we note that the performance on XM2VTS is comparable to ours, while on the BioID their results are significantly better.

Other works face the same problem, while adopting a different metrics. For instance [Wang et al., 2005] adopt a normalized mean error (not the maximum) and give an error of 2.67% on the entire FRGC. By adopting this measure on the considered FRGC subsets we observe an error of 3.21%. Analogously, [Fasel et al., 2005] provide the localization results on the BioID in terms of the mean relative error, this time expressed in iris units. Noting that the iris diameter is slightly shorter than the 20% of the inter-ocular distance, their measurement corresponds to a mean error (relative to the inter-ocular distance) of 0.04, while we report a mean relative error of 0.031. The method described by [Everingham and Zisserman, 2006] carries out the experiments on the FERET database: in the 90% of images the mean relative error is reported to be smaller or equal to 0.047, which is remarkable (for the same level of precision, on the FERET we count about the 81% of images).

We also present in Figure 10 the histograms of Δ_x, Δ_y, Δ_s, Δ_α (recall Sec. 3.) made by our eye localization module on all the datasets previously considered; for comparison, we report in Figure 11 the results of the CLM algorithm on the available datasets (BioID, XM2VTS).

Referring to the FR algorithm DCT/GMM proposed by [Rodriguez et al., 2006], we observe that each error histogram generated by the coarse-to-fine technique is entirely included within the declared error tolerance (rotation error $\in$ [-10°, 10°], translational error $\in$ [-0.2, 0.2], scale error $\in$ [0.8, 1.2]). In the spirit of their article, we conclude that our application would be appropriate for the initialization of DCT/GMM.

The speed was not the main focus of our research, giving that nowadays there exist dedicated architectures which would allow to obtain a real-time application. Running java interpreted code on a Pentium 4 with 3.2GHz, we report the computational time of the two

[10] The authors attribute this behavior to the major similarity of BioID images to the images used to train CLM.

modules: on average eye detection requires about 4 seconds on faces with an inter-ocular distance of 70 pixels, while eye localization takes about 12 seconds.
We have investigated the possibility of reducing the cardinality of the SVM2. As already pointed out, the entity of the support vectors reduction is proportional to the threshold imposed on the maximum marginal difference; in particular we have carried out the experiments by fixing the threshold at 0.5 and 1. The value 0.5 is chosen to interpolate between 0 and 1 in order to sketch the trend of the performance reduction vs. the SV reduction.

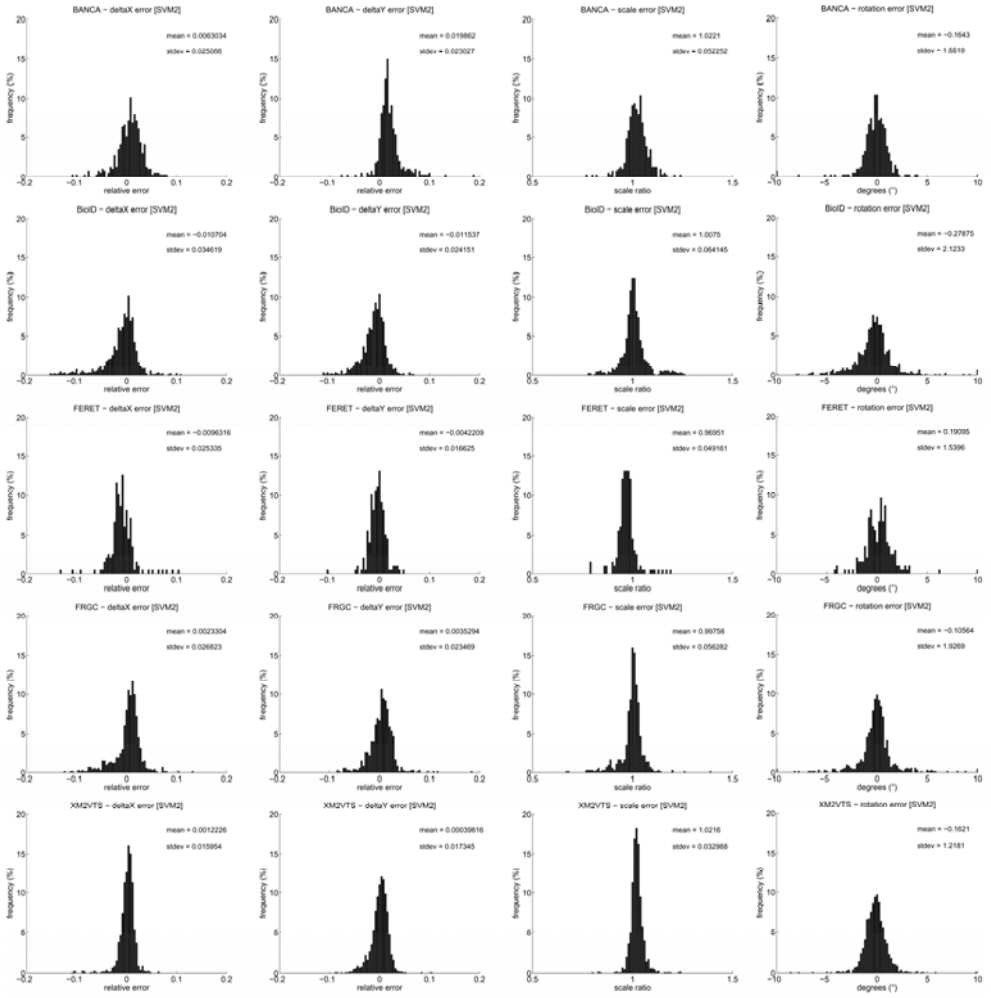

Figure 10. The histograms of the horizontal, vertical, scale and rotation error of the eye localization module (SVM2)

Thresholds 1 and 0.5 led respectively to a reduction of the original SVM2 from 3209 SVs to 529 and 1716. As the computational cost of the eye locator is three times bigger than that of the eye detector, and as it is linearly dependent on the number of SVs, these reductions

roughly correspond to a global application speed-up of 60% and 35% respectively. There is a clear trade-off between the entity of the reduction and the accuracy of the localization: the performance of the localization module, measured on a randomly chosen subset (400 images) of the XM2VTS, and expressed in terms of AUC, decreased by about 3.3% and 0.6% respectively (See graph 12.). This is quite a good result, especially regarding the latter experiment. On the other hand, if this deterioration of the localization precision is not acceptable for a certain face processing application, then the original SVM2 should be used instead.

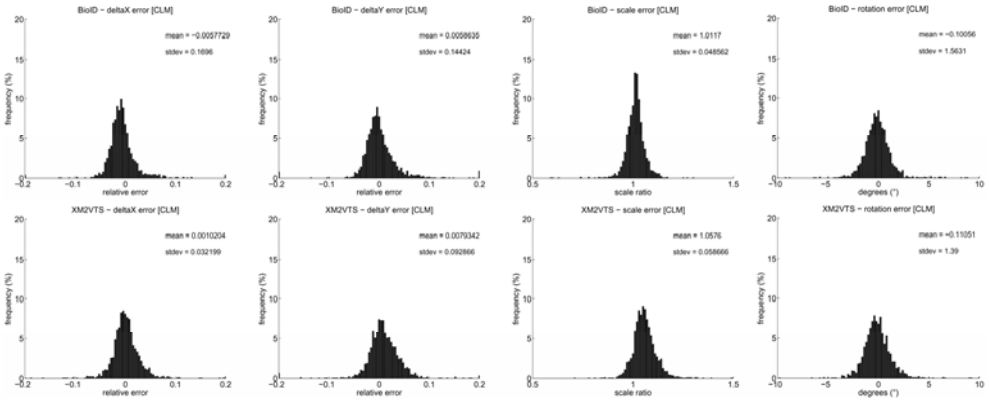

Figure 11. The histograms of the horizontal, vertical, scale and rotation error of the CLM algorithm

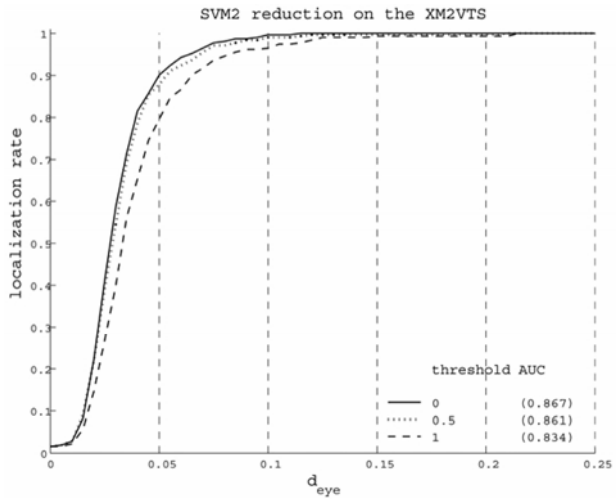

Figure 12. Support vectors reduction experiment

5. From eye centers to fiducial points

In this section we show how, given the eye centers, we derive a set of 27 characteristic points (*fiducial points*): three points on each eyebrow, the tip, the lateral extremes and the vertical mid-point of the nose, the eye and lip corners, their upper and lower mid-points, the mid-point between the two eyes, and four points on the cheeks (see Figure 13).
This module has been conceived to work on still color images of good quality, acquired with uniform illumination, where the face is almost frontal and the subject assumes either a neutral or a slightly smiling expression.
The method proceeds in a top-down fashion: given the eye centers, it derives the eye, nose and mouth subimages on the basis of simple geometrical considerations, and extracts the corresponding fiducial points (green points in Figure 13) as described in the following. Finally, in order to enrich the face description, further fiducial points (red points in Figure 13) are inferred on the basis of the position of the extracted points.

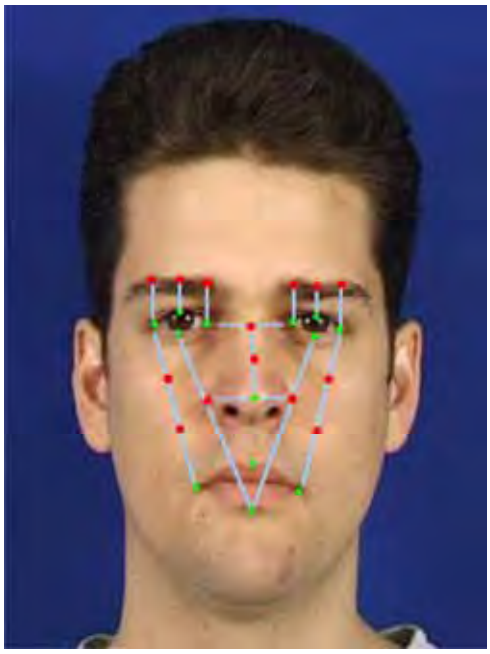

Figure 13. A face is described by 27 fiducial points: 13 are directly extracted from the image (in green), 14 are inferred from the former ones (in red)

5.1 Eyes

The eyes are described by a parametric model which is a simplified version (6 parameters instead of 11) of the deformable template proposed in [Yuille et al., 1992].
The eye model is made of two parabolas, representing the upper and lower eye arcs, and intersecting at the eye corners (see Figure 14); the model parameters, $\vec{p} = \{x_t, y_t, a, b, c, \theta_t\}$, are: the model eye center coordinates (x_t, y_t), the eye upper and lower half-heights a and c,

the eye half-width b, and the rotation angle θ_t expressing the rotation of the model with respect to the horizontal axis.
The fundamental step to obtain good results is a very precise initialization of the template parameters. To this end, the eye center coordinates, (x_c, y_c), derived by the SVM2, are used as initial values for (x_t, y_t). In order to find a good initial estimate for the parameters a, b, c, we carried out a statistical study on 2000 images to evaluate the relation between the inter-ocular distance d and both the semi-width, b and the semi-height of the eye, a and c, obtaining very stable results: the mean values are 5.6 and 12 respectively, with small variance values (0.8 and 1.2), making these evaluations reliable and useful to set the initial values of the parameters a, b, c correspondingly. The last parameter, θ, is set initially to the estimated face tilt.

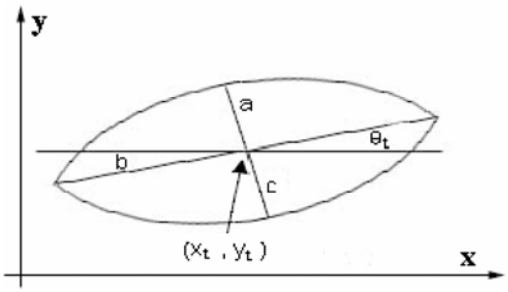

Figure 14. Deformable eye template

In order to adapt the generic template to a specific eye, we minimize an energy function E_t that depends on the template parameters (prior information on the eye shape) and on certain image characteristics (edges and the eye sclera). The characteristics are evaluated on the u plane of the CIE-Luv[11] space, since in this color plane the information we are looking for (edges and eye sclera) are strengthened and clearer (see Figure 15 b,c). More precisely:

$$E_t = E_{prior} + E_e + E_i,$$

where:

1. $E_{prior} = \frac{k_1}{2}\left((x_t - x_c)^2 + (y_t - y_c)^2\right) + \frac{k_2}{2} \cdot (b - d/12)^2 + \frac{k_3}{2}\left((b - 2a)^2 + (a - 2c)^2\right)$
2. $E_e = -\frac{c_1}{|\partial R_w|} \cdot \int_{\partial R_w} \Phi_e(\vec{x}) ds$,
   being ∂R_w the upper and lower parabolas, and Φ_e the edge image obtained applying the Sobel filter to the eye subimage.
3. $E_i = -c_2 \int_{R_w} \Phi_i(\vec{x}) ds$,
   where R_w is the region enclosed between the two parabolas, and Φ_i is a weighted image called *eye map*, and determined as follows:
   - threshold the u plane with a global threshold:
     $th = 0.9 \times \max(u)$

[11] Uniform color space introduced by the CIE (Commission Internationale de l'Eclairage) to properly represent distances between colors [Wyszecki and Stiles, 1982].

- adapt the threshold until the pixels set to 1 are symmetrically distributed around the eye center.
- for every pixel p

$$\Phi_i(p) = \begin{cases} 255 & \text{if p is white} \\ -100 & \text{if p is black} \end{cases}$$

The function is optimized adopting a search strategy based on the steepest descent, as suggested in Yuille's work; once obtained the eye contour description, we derive the two eye corners and the upper and lower mid-points straightforwardly (see Figure 15).

5.2 Nose

The nose is characterized by very simple and generic properties: the nose has a "base" the gray levels of which contrast significantly with the neighboring regions; moreover, the nose profile can be characterized as the set of points with the highest symmetry and high luminance values; therefore we can identify the nose tip as the point that lies on the nose profile, above the nose baseline, and that corresponds to the brightest gray level. These considerations allow to localize the nose tip robustly (see Figure 16).

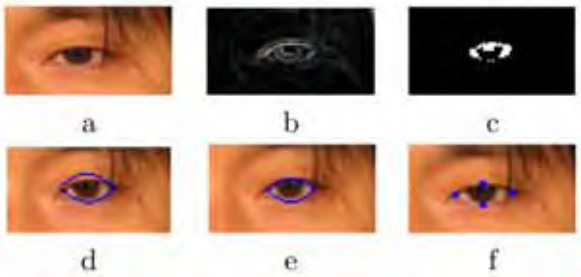

Figure 15. Eye points search: a) eye subimage b) edge image c) eye map d) initial template position e) final template position f) fiducial points

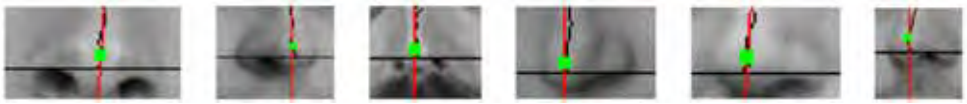

Figure 16. Examples of nose processing. The black horizontal line indicates the nose base; the black dots along the nose are the points of maximal symmetry along each row; the red line is the vertical axis approximating those points; the green marker indicates the nose tip

5.3 Mouth

Regarding the mouth, our goal is to locate its corners and its upper and lower mid-points. To this aim, we use a snake [Hamarneh, 2000] to determine the entire contour since we verified that they can robustly describe the very different shapes that mouths can assume.
To make the snake converge, its initialization is fundamental; therefore the algorithm estimates the mouth corners and anchors the snake to them: first, we represent the mouth subimage in the *YCbCr* color space, and we apply the following transformation:

$$MM = (255 - (C_r - C_b))\, C_r^2$$

MM is a mouth map that highlights the region corresponding to the lips; *MM* is then binarized putting to 1 the 20% of its highest values; the mouth corners are determined taking the most lateral extremes (see Figure 17).

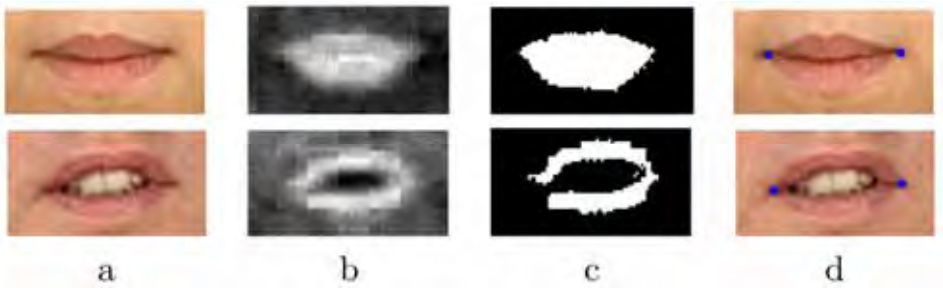

Figure 17. Mouth corners estimation: a) mouth subimage b) mouth map c) binarized mouth map d) mouth corners

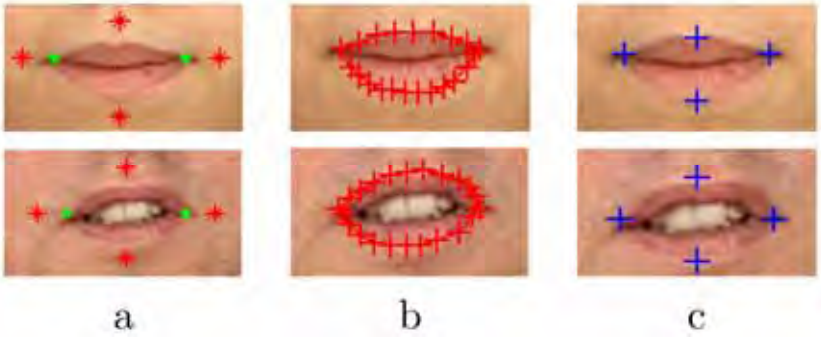

Figure 18. Snake evolution: a) snake initialization b) final snake position c) mouth fiducial points

The snake we used to find the mouth contour is composed of an initial set *S* of 4 points: the mouth corners and 2 points taken as a function of both the mouth subimage dimensions and of the mouth corner positions (see Figure 18 a). To better describe the contour, the size of *S* is automatically increased, while the snake is being deformed, by adding points where the contour presents high curvature values.
In order to deform the snake, a force F_{tot} is applied to each point $P = (x, y) \in S$:

$$F_{tot}(P) = aF_{ext}(P) + bT_F(P) + cF_F(P) + dI_F(P)$$

It is constituted of both external and internal forces. F_{ext} is external and deforms the snake in order to attract it to the mouth contour extracted from *MM*

$$\begin{aligned} F_{ext}(P(x,y)) \quad = \quad & \tfrac{1}{2}\left[\|\nabla MM(x,y+1)\| - \|\nabla MM(x,y-1)\|\right., \\ & \left.\|\nabla MM(x+1,y)\| - \|\nabla MM(x-1,y)\|\right] \end{aligned}$$

while T_F , F_F , I_F are internal forces that constrain the snake to stay continuous and smooth

$$\begin{aligned} T_F(P(x,y)) &= \left[\frac{\partial^2 P}{\partial x^2}, \frac{\partial^2 P}{\partial y^2}\right] \\ F_F(P(x,y)) &= \left[\frac{\partial^2 T_F(P)}{\partial x^2}, \frac{\partial^2 T_F(P)}{\partial y^2}\right] \\ I_F(P(x,y)) &= \vec{n}(x,y) \end{aligned}$$

where $\vec{n}(x,y)$ is the vector in $P(x, y)$ normal to the snake.
The algorithm adds points and deforms the snake until the global force F_{tot} is lower than a certain tolerance for a fixed number of consequent steps. Once obtained the mouth contour description, we derive the fiducial points straightforwardly. Figure 18 reports some results; we notice that the described method achieves good results both on closed and open mouths.

5.4 Evaluation of the fiducial points precision

In order to quantitatively evaluate the precision of the extracted fiducial points (FP), we adopt the error measure d_{FP} that can be considered an extension of d_{eye} to a bigger set of features

$$d_{FP} = \frac{1}{|FP|} \sum_{P \in FP} \frac{\|P - \tilde{P}\|}{\|C_l - C_r\|}$$

where $\tilde{P}$ is the localized position of a fiducial point and P is the corresponding ground truth. Notice that d_{FP} is a statistics different from d_{eye} as it averages the localization errors instead of taking their maximum. On one hand this is a less demanding criterion, however it is a more representative measure of a larger set of features.
Unfortunately, such performance evaluation is rarely given in the related literature. As we have been provided with the localization output of the CLM method on the XM2VTS database, we are able to compare it with our own. On the 9 fiducial points that are common to both methods (eye corners, nose tip, mouth corners and mid-points), we obtain a d_{FP} equal to 0.051 while CLM achieves 0.056. Regarding solely our method, if we take into account also the 4 eye mid-points, the precision considerably improves to 0.045. The remaining 14 fiducial points are not considered for the performance evaluation because they are inferred from the other 13 and their precision is correlated.
Furthermore, a disjoint analysis of the precision achieved over each fiducial point highlights that the nose tip is the most critical one (mean error of 0.07), while the points lying around the eyes are the most precisely determined (mean error of 0.03).

6. Face recognition experiment

We set up a simple face recognition experiment to investigate the behavior of two different FRTs when initialized on real outputs of our feature extraction method. The techniques, LAIV and CAS, have been chosen in such a way to represent two different processing paradigms: the former is based on local features, the latter treats the information at the global face level. For this experiment we do not consider any more the CSU baseline methods considered in Sec. 3. since they are not state-of-the-art FRTs, being their purpose only comparative. Instead, LAIV and CAS are very recent methods which are reported to score high recognition rates.

The experiment has two aims: to compare the absolute performance achieved by either method; to analyze the relative performance decay of each FRT in function of the eye localization precision.

LAIV-FRT: This technique is a feature-based FRT described in [Arca et al., 2006]. Given the eye positions, it uses the technique described in Sec. 5. to automatically locate the position of 27 fiducial points. Each fiducial point is characterized by extracting square patches centered in them and convolving those with the Gabor filter bank described in [Wiskott et al., 1999]. The resulting 40 coefficients are complex numbers, and the jet J is obtained by considering only the magnitude part. Thus, the face characterization consists of a *jets vector* of 40 × 27 real coefficients.

The recognition task becomes the problem of finding a suitable similarity measure between jets. The LAIV technique introduces the idea of considering only the set of points for which the corresponding jets have high similarity. In particular, to recognize a test image t, it is compared one-to-one with each image i belonging to the gallery G, producing a similarity score, and it is recognized as the subject i^* which obtained the highest score:

- for each image $i \in G$ and each fiducial point k = 0, .., 26, compute the similarity measure between pairs of corresponding Jets:

$$S^{i,k} = S(J^{t,k}, J^{i,k}) = \frac{\sum_z J_z^{t,k} J_z^{i,k}}{\sqrt{\sum_z (J_z^{t,k})^2 \sum_z (J_z^{i,k})^2}}$$

  where z = 0, ..., 39 and $J_{t,k}$ is the Jet in the test image corresponding to the k^{th} fiducial point.
- for each fiducial point k, order the values $\{S^{i,k}\}$ in descending order, and assign to each of them a weight $w^{i,k}$ as a function of its ordered position $p^{i,k}$:

$$w^{i,k} = c \cdot [\ln(x + y) - \ln(x + p^{i,k})],$$

  where $y = \frac{|G|}{4}$, $x = e^{-\frac{1}{2}}$, and c is a normalization factor.
- for each gallery image i, the similarity score is obtained as a weighted average of the pairwise jet similarity, limited to the set *BestPoints* of $\left\lfloor \frac{27}{2} \right\rfloor$+ 1 = 14 points with highest weight:

$$\text{score}(i) = \sum_{k \in BestPoints} w^{i,k} S^{i,k}.$$

This technique gives better results than considering the average of all similarities, since it allows to discard wrong matches on single points: if some fiducial points are not precisely localized either in the test or in the gallery image, they will have low similarity measures and will not belong to the set *BestPoints*, so they will not be used for recognition.

CAS-FRT: We consider here a custom reimplementation of the method proposed by [Zhang et al., 2005]; the authors have successively developed the technique in [Shan et al., 2006], which however requires an extremely long learning phase.

Just like LAIV-FRT, CAS does not need any training procedure to construct the face model. First it proceeds to normalize each face to a size of 80×88 pixels, obtained by means of an affine transformation of the original image so that the eye centers are brought in predefined positions and their distance is 40 pixels. The knowledge of the eye locations is sufficient to compute this transformation.
Secondly, a multi-scale face representation is obtained by convolving the normalized face with the same bank of 40 Gabor filters as before, this time computed pixelwise on the whole face; the result is a set of 40 *Gabor magnitude pictures* (GMPs). Since the Gabor magnitude changes very slowly with the displacement, the information in the GMPs is further enhanced by applying the local binary pattern (LBP) operator [Ojala et al., 2002], to obtain 40 *local Gabor binary pattern maps* (LGBP maps). Each LGBP map is spatially divided into non-overlapping regions (with a 4×8 pixel size), then the histograms of all regions are computed and concatenated in a *histogram sequence* (LGBPHS) that models the face (see Figure 19 for a visual representation of the whole procedure).

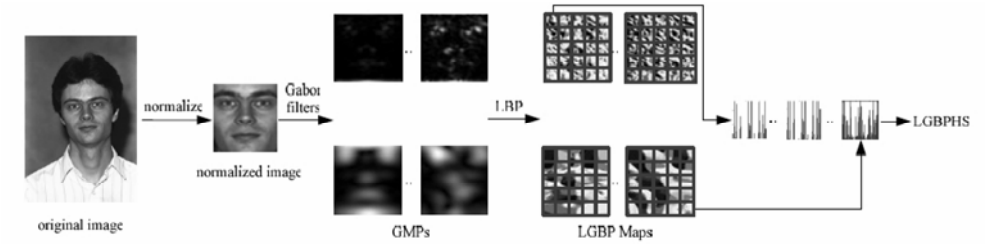

Figure 19. The face pre-processing of CAS-FRT

Finally, the technique of histogram intersection is used to measure the similarity between different face models to achieve face recognition.

Analogously to what done in Sec. 3., the recognition experiments are carried out on the XM2VTS. However, as both LAIV-FRT and CAS-FRT need no training, now it is possible to use all sessions but one (used as gallery) as probe set.
Table 1. reports the face recognition rate of LAIV-FRT and CAS-FRT when initialized respectively on the eye ground truth positions, and on the localization output by the eye detector and locator.

Initialization	FR rate	
	LAIV-FRT	CAS-FRT
ground truth	95.1%	96.4%
eye detector	92.3%	82.8%
eye locator	93.5%	87.9%

Table 1. The face recognition rate of LAIV-FRT and CAS-FRT with different initializations

It can be noted that CAS-FRT performs better than LAIV-FRT (96.4% vs. 95.1%) when it is manually and precisely initialized, but its performance drops dramatically when an automatic eye localization method is used. On the contrary, LAIV-FRT proves to be more robust with respect to localization errors; indeed, it can overcome slight mis-initializations. It can be stated that LAIV-FRT behaves globally better than CAS-FRT as it is more robust in the spirit of Eq. (1).
This difference in performance is probably due to the global nature of CAS initialization: if the eye centers estimation is mistaken, the error will propagate to the rest of the face regions due to the global affine transformation. Also in the case of LAIV-FRT the error affects the computation, but in a more local sense: first of all, this FRT relies on the measured inter-ocular distance to scale the Gabor filters, however the histogram of the scale error is quite narrow (see the third graph of the last row of Figure 10); secondly, a slightly wrong initialization of the employed templates is often recovered by the template matching algorithms. Anyways, even when a full recovery is not attained, the selection criterion of the *BestPoints* set allows to discard the unreliable fiducial points and LAIV-FRT still manages to recognize the face in a number of cases. On the other hand, it should be observed that the presence of the intermediate module described in Sec. 5., and the discard operated by the selection criterion, weaken the dependency between the eye localization precision and the recognition rate, so that the performance results on the different initializations are very similar.
The same phenomenon explains the results of the experiment reported in Figure 2 regarding artificially perturbed manual annotations: all the considered CSU face recognition techniques start from a global representation of the face and hence are greatly affected by misalignments.

7. Conclusion

The subject of this chapter is the presentation of a novel method for the automatic and precise localization of facial features in 2D still images. The method follows the top-down paradigm and consists of subsequent steps to decompose the initial problem in increasingly easier tasks: assuming a rough localization of the face in the image, first comes the application of an eye detector with the aim of discriminating between real face regions and possible false positives. The accuracy of the detection is nearly optimal. Successively, an eye locator is applied on a small neighborhood of the detector output to improve the localization precision. Finally, the eye center positions are used to derive 27 facial fiducial points, either extracted directly from the image or inferred on the basis of simple geometrical considerations.
The eye localization module has been extensively tested on five publicly available databases with different characteristics to remark its generality. In the overall, the results are comparable to or better than those obtained by the state-of-the-art methods. The performance evaluation is carried out according to two objective performance measures in order to favor the comparison with other techniques. Concerning the fiducial point localization, results on the XM2VTS show high precision.
In the last years many research works have pointed out the importance of facial feature localization as the fundamental step for the initialization of other methods, mostly face recognition techniques. In general, not all types of error affect the subsequent processing in the same way: for instance scale errors usually affect a FR technique more than translational

misalignment. Moreover, face recognition techniques manifest a different tolerance to the localization error depending on the nature of their initialization. We conducted some experiments which suggest that, as the localization precision decreases, the recognition rate decays more rapidly for those methods which start from a global face representation. However, since different FR techniques exhibit a different robustness to certain types and amount of error, there exists no absolute threshold for precise localization. The authors of face recognition techniques should investigate the robustness of their methods with respect to misalignments, in order to state the error tolerance that they assume when declaring the face recognition rate.
Both the obtained localization results and the survey of recent eye localization techniques clearly show that we are far from perfect localization and there is still room for improvement.

8. Appendix: datasets

This appendix details the definition of the considered public databases, specifying for each of them which images have been used to carry out the experimental tests. In alphabetical order:

- The [BANCA DB, web] of English people consists of three sections referred to as Controlled, Adverse and Degraded. The latter is not considered as the images are particularly blurred, making the step of precise eye localization useless. Regarding the former:
  - **Controlled**: it consists of 2080 images each one representing one person placed in front of the camera and standing on a uniform background. The database collects pictures of 52 people of different ethnic groups (Caucasian, Indians, Japanese, Africans, South-Americans), acquired in 4 different sessions (10 images per subject in each session). The illumination conditions vary from daylight to underexposed, while no evident chromatic alteration is present.
  - **Adverse**: like the **Controlled** section it consists of 2080 images, each one representing one person placed in front of the camera and looking down as if reading, while in this section the background is non-uniform. The image quality and illumination are not very good.

  The selected test set is composed of the first image of each subject in each section, for a total of 416 images.
- The [BioID DB, web] is formed of 1521 gray scale images of close-up faces. The number of images per subject is variable, as it is the background (usually cluttered like in an office environment).
  The tests reported in the previous sections refer to the whole database.
- The [FERET DB, web] database consists of 10 gray level images per person organized according to the out of plane rotation: 0°, ±15°, ±25°, ±40° or ±60°; regarding the sole frontal views the set contains two images per subject, one smiling, one with neutral expression.
  The considered test set consists of 1000 images randomly selected from the images with rotation up to ±15°.

- The [FRGC DB, web] database version 1.0 collects 5658 high resolution images of 275 subjects in frontal position, arranged in two sections: controlled and uncontrolled. The images are organized in subject sessions: each contains 4 images acquired in controlled conditions (uniform background and homogeneous illumination) and 2 in uncontrolled conditions (generic background and varying illumination conditions). In both conditions, half of the images represent the subject while smiling, the remaining half with neutral expression. The number of sessions varies from subject to subject, between 1 and 7.
  The considered test set is composed of both 473 controlled and 396 uncontrolled images. These numbers are obtained by taking, for each subject, the first controlled image of the first two sessions (when the second is present).
- The [XM2VTS DB, web] consists of 1180 high quality images of single faces acquired in frontal position and with homogeneous background; some of the subjects wear spectacles. The pictures are grouped into 4 sessions of 295 subjects each.
  The conducted tests refer to the whole database.

9. References

Arca, S., Campadelli, P., and Lanzarotti, R. (2006). A face recognition system based on automatically determined facial fiducial points. *Pattern Recognition*, 39(3):432–443. [Arca et al., 2006]

BANCA DB (web). Address: http://www.ee.surrey.ac.uk/Research/VSSP/banca/.[BANCA DB, web]

Belhumeur, P., Hespanha, J., and Kriegman, D. (1997). Eigenfaces vs. Fisherfaces: Recognition using class specific linear projection. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 19(7):711–720. [Belhumeur et al., 1997]

Beveridge, J., Bolme, D., Draper, B., and Teixeira, M. (2005). The CSU face identification evaluation system. its purpose, features, and structure. *Machine vision and applications*, 16:128–138. [Beveridge et al., 2005]

BioID DB (web). Address: http://www.humanscan.de/support/downloads/facedb.php. [BioID DB, web]

Burges, C. (1996). Simplified Support Vector decision rules. *Int'l Conf. Machine Learning*, pages 71–77. [Burges, 1996]

Campadelli, P., Lanzarotti, R., and Lipori, G. (2005). Face localization in color images with complex background. *Proc. IEEE Int'l Workshop on Computer Architecture for Machine Perception*, pages 243–248. [Campadelli et al., 2005]

Campadelli, P., Lanzarotti, R., and Lipori, G. (2006a). Eye localization and face recognition. RAIRO - *Theoretical Informatics and Applications*, 40:123–139. [Campadelli et al., 2006a]

Campadelli, P., Lanzarotti, R., and Lipori, G. (2006b). Precise eye localization through a general-to-specific model definition. *Proceedings of the British Machine Vision Conference*, 1:187–196. [Campadelli et al., 2006b]

Cristinacce, D. and Cootes, T. (2006). Feature detection and tracking with constrained local models. *Proc. the British Machine Vision Conf.*, 3:929–938. [Cristinacce and Cootes, 2006]

Everingham, M. and Zisserman, A. (2006). Regression and classification approaches to eye localization in face images. *Proceedings of the 7th International Conference on Automatic Face and Gesture Recognition* (FG2006), pages 441–446. [Everingham and Zisserman, 2006]

Fasel, I., Fortenberry, B., and Movellan, J. (2005). A generative framework for real time object detection and classification. *Computer Vision and Image Understanding*, 98:182–210. [Fasel et al., 2005]

FERET DB (web). Address: http://www.itl.nist.gov/iad/humanid/feret/.[FERET DB, web]

FRGC DB (web). Address: http://www.frvt.org/FRGC/.[FRGC DB, web]

Gizatdinova, Y. and Surakka, V. (2006). Feature-based detection of facial landmarks from neutral and expressive facial images. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 28(1):135–139. [Gizatdinova and Surakka, 2006]

Hamarneh, G. (2000). Image segmentation with constrained snakes. *Swedish Image Analysis Society Newsletter* SSABlaskan, 8:5–6. [Hamarneh, 2000]

Hamouz, M., Kittler, J., Kamarainen, J., Paalanen, P., Kälviäinen, H., and Matas, J. (2005). Feature-based affine invariant localization of faces. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 27(9):1490–1495. [Hamouz et al., 2005]

Huang, J. and Wechsler, H. (1999). Eye detection using optimal wavelet packets and radial basis functions (RBFs). *Int'l Journal of Pattern Recognition and Artificial Intelligence*, 13(7):1009–1026. [Huang and Wechsler, 1999]

Jesorsky, O., Kirchberg, K., and Frischholz, R. (2001). Robust face detection using the Hausdorff distance. *Lecture Notes in Computer Science*, 2091:212 – 227. [Jesorsky et al., 2001]

Ji, Q., Wechsler, H., Duchowski, A., and Flickner, M. (2005). Special issue: eye detection and tracking. *Computer Vision and Image Understanding*, 98(1):1–3. [Ji et al., 2005]

Ma, Y., Ding, X., Wang, Z., and Wang, N. (2004a). Robust precise eye location under probabilistic framework. *Proc. IEEE Int'l Conf. Automatic Face and Gesture Recognition*, pages 339–344. [Ma et al., 2004a]

Ma, Y., Ding, X., Wang, Z., and Wang, N. (2004b). Robust precise eye location under probabilistic framework. *Proc. IEEE Int'l Conf. Automatic Face and Gesture Recognition*. [Ma et al., 2004b]

Martinez, A. (2002). Recognizing imprecisely localized, partially occluded, and expression variant faces from a single sample per class. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 24(6):748–763. [Martinez, 2002]

Min, J., Bowyer, K. W., and Flynn, P. J. (2005). Eye perturbation approach for robust recognition of inaccurately aligned faces. *Proc. of the Internationl Conf. Audio and Video based Biometric Person Authentication* (AVBPA), LCNS 3546:41–50. [Min et al., 2005]

Nguyen, D. and Ho, T. (2005). An efficient method for simplifying Support Vector Machines. *Proc. Int'l Conf. Machine learning*, pages 617–624. [Nguyen and Ho, 2005]

Niu, Z., Shan, S., Yan, S., Chen, X., and Gao, W. (2006). 2D Cascaded AdaBoost for Eye Localization. *Proc. of the 18th Internationl Conference on Pattern Recognition*, 2:1216–1219. [Niu et al., 2006]

Ojala, T., Pietikäinen, M., and Mäenpää, T. (2002). Multiresolution grayscale and rotation invariant texture classification with local binary patterns. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(7):971–987. [Ojala et al., 2002]

Oren, M., Papageorgiou, C., Sinha, P., Osuna, E., and Poggio, T. (1997). Pedestrian detection using wavelet templates. *In Proc. Computer Vision and Pattern Recognition*, pages 193–199. [Oren et al., 1997]

Osadchy, M., Miller, M., and LeCun, Y. (2005). Synergistic face detection and pose estimation with energy-based models. In Saul, L. K., Weiss, Y., and Bottou, L., editors, *Advances in Neural Information Processing Systems*, volume 17, pages 1017–1024. MIT Press. [Osadchy et al., 2005]

Rodriguez, Y., Cardinaux, F., Bengio, S., and Mariéthoz, J. (2006). Measuring the performance of face localization systems. *Image and Vision Computing*, 24:882– 893. [Rodriguez et al., 2006]

Schneiderman and Kanade, T. (2004). Object detection using the statistics of parts. *Int'l Journal of Computer Vision*, 56(1):151–177. [Schneiderman and Kanade, 2004]

Schölkopf, B., Mika, S., Burges, C., Knirsch, P., Muller, K., Rätsch, G., and Smola, A. (1999). Input space vs. feature space in kernel-based methods. *IEEE Trans. Neural Networks*, 10(5):1000–1017. [Schölkopf et al., 1999]

Shakhnarovich, G. and Moghaddam, B. (2004). Face recognition in subspaces. In *Handbook of Face Recognition*, Springer-Verlag. [Shakhnarovich and Moghaddam, 2004]

Shan, S., Chang, Y., Gao, W., and Cao, B. (2004). Curse of mis-alignment in face recognition: problem and a novel mis-alignment learning solution. *Int'l Conf. Automatic Face and Gesture Recognition*, pages 314–320. [Shan et al., 2004]

Shan, S., Zhang, W., Y.Su, Chen, X., and Gao, W. (2006). Ensemble of Piecewise FDA Based on Spatial Histograms of Local (Gabor) Binary Patterns for Face Recognition. *IEEE Proc. the 18th Int'l Conf. Pattern Recognition*, ICPR 2006 Hong Kong. [Shan et al., 2006]

Song, J., Chi, Z., and Liu, J. (2006). A robust eye detection method using combined binary edge and intensity information. *Pattern Recognition*, 39(6):1110–1125. [Song et al., 2006]

Tang, X., Ou, Z., Su, T., Sun, H., and Zhao, P. (2005). Robust Precise Eye Location by AdaBoost and SVM Techniques. *Proc. Int'l Symposium on Neural Networks*, pages 93–98. [Tang et al., 2005]

Vapnik (1995). *The nature of statistical learning theory*. Springer. [Vapnik, 1995]

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

Viola, P. and Jones, M. (2004). Robust real time object detection. *Int'l Journal of Computer Vision*, 57(2):137–154. [Viola and Jones, 2004]

Wang, P., Green, M., Ji, Q., and Wayman, J. (2005). Automatic eye detection and its validation. *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 3:164ff. [Wang et al., 2005]

Wiskott, L., Fellous, J., Kruger, N., and von der Malsburg, C. (1999). *Face recognition by elastic bunch graph matching*. pages 355–396. CRC Press. [Wiskott et al., 1999]

Wyszecki, G. and Stiles, W. (1982). *Color science: concepts and methods, quantitative data and formulae*. John Wiley and Sons, New York, N.Y. [Wyszecki and Stiles, 1982]

XM2VTS DB (web). Address:
http://www.ee.surrey.ac.uk/Research/VSSP/xm2vtsdb/.[XM2VTS DB, web]

Yuille, A., Hallinan, P., and Cohen, D. (1992). Feature extraction from faces using deformable templates. *Int'l journal of computer vision*, 8(2):99–111. [Yuille et al., 1992]

Zhang, W., Shan, S., Gao,W., Chen, X., and Zhang, H. (2005). Local Gabor Binary Pattern Histogram Sequence (LGBPHS): A Novel Non-Statistical Model for Face Representation and Recognition. *Proceedings of the Tenth IEEE International Conference on Computer Vision* (ICCV), Beijing, China, pages 786 – 791. [Zhang et al., 2005]

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

Zhou, Z. and Geng, X. (2004). Projection functions for eye detection. *Pattern Recognition Journal*, 37:1049–1056. [Zhou and Geng, 2004]

Zhu, Z. and Ji, Q. (2005). Robust real-time eye detection and tracking under variable lighting conditions and various face orientations. *Computer Vision and Image Understanding*, 98:124–154. [Zhu and Ji, 2005]