LOCALIZATION OF FACIAL FEATURES AND FIDUCIAL POINTS

Paola Campadelli
Dipartimento di Scienze dell'Informazione
Università degli Studi di Milano
Via Comelico, 39/41 20135 Milano, Italy
email: campadelli@dsi.unimi.it

Raffaella Lanzarotti
Dipartimento di Scienze dell'Informazione
Università degli Studi di Milano
Via Comelico, 39/41 20135 Milano, Italy
email: lanzarotti@dsi.unimi.it

Abstract

In this paper we present an algorithm for the automatic detection and description of facial features in 2D color images of either frontal or rotated human faces. The algorithm first identifies the sub-images containing each feature, afterwards, it processes them separately to extract the characteristic fiducial points. The features are looked for in down-sampled images, the fiducial points are identified in the high resolution ones. The method uses both color and shape information and does not require any manual setting or operator intervention.

Keywords: Pattern Analysis and Recognition, Facial Feature Detection, Face Recognition, Image Processing.

1 Introduction

The ability to localize and describe precisely facial features (eyes, nose and lips) in images are important tasks for applications like face recognition [1], model-based coding of video sequences [2], 3D face reconstruction [3], and intelligent man-machine interfaces [2]. Such is not an easy machine-vision task due to the high interpersonal variability (gender, race, ...), the intra-personal changes (pose, expression, ...), and acquisition conditions (lighting, image resolution, ...).

The algorithms reported in literature can be classified into *color-based* and *shape-based.* The first class of methods characterizes the face and each feature with a certain combination of colors [4]. This is a low-cost approach, but, not very robust. The shape-based approaches look for specific shapes in the image adopting either template matching (with deformable templates [5] or not [6]), graph matching [7], snakes [8], or the Hough transform [9]. Although these methods give good results, they are computationally expensive and they often work only under restricted assumptions (regarding the head position and the illumination conditions).

In this paper we describe a technique which uses both color and shape information to automatically identify a set of feature fiducial points with great reliability. Results on a database of 200 color images, taken at different orientations, illumination conditions and resolution are reported and discussed.

2 Method overview

The method we propose works on images of face' foregrounds. We acquire color images with homogeneous and light-colored background, and frontal and diffuse illumination. The images have a wide variety of resolutions. Faces can be either in frontal position or rotated around the head vertical axis of 30° at most, moreover the method is robust to little lateral head tilts of about 10°. In any case, the completeness of the facial images is requested: no occlusions, no wearing glasses, no beard and closed mouth.

The algorithm has two hierarchical processing modules: the first identifies four sub-images, each tightly containing one of the features of interest; the latter module is specialized in localizing fiducial points on the found features with high accuracy.

A preliminary solution to this problem, experimented on a low number of images, has been presented in [10]. Now we have modified completely the module for localizing the sub-images making it more general, more robust and efficient. Moreover we have partially changed the method for localizing the fiducial points.

3 Identification of sub-images

In this module the images are down-sampled to a resolution in the range between 150x170 to 300x340 pixels.

At first, we divide the features of interest from the skin and the background, by clustering the grey level image into three clusters through the clustering algorithm presented in [11]. The lightest grey level represents the background, the intermediate the skin and the darkest represents both the features and other dark pixels of the image (for example the hair) [Fig.1].

To localize the features of interest the largest region, S, with pixels of the intermediate grey-level (corresponding to the skin) is found. Then all the pixels surrounded by pixels of S and belonging to the darkest grey-level are identified and set to 1; all the others are set to 0. What we obtain is the *feature image* [Fig.2].

We observe that, besides the features of interest, also few pixels, such as those corresponding to shadows or to the hair, are set to 1. Further processing is therefore required to isolate the features of interest in separated sub-images.

We proceed localizing the eyes, then, in order to

Figure 1. Clustered images

Figure 2. Features images

look for the lips and the nose, the attention can be concentrated upon a restricted image area.

3.1 Eyes localization

In order to determine the set of rows (*eyes band*) which contains the eyes, we apply the template matching to the *features image* [Fig.2], searching the two eyes. The difficulty is that we are not looking for an object with a fixed shape [Fig.3]. For this reason we adopt a binary template which models the two eyes in a very rough way. It consists of two symmetric blobs placed side by side each being large enough to overlap to the region corresponding to an eye in the *features image*. A single template [Fig.4] has been used for all the images which are of significant different size, thus showing a desirable scale-independence property.

Among the positions with the highest cross-correlation, we maintain the 10 which satisfy also the following symmetry condition: the cross correlation between half the template (one blob) and the sub-images on the left and on the right of the found position are compared. The position is rejected if the results are not similar enough, that is their ratio is lower than 0.7.

On all the images of our database most of the points calculated in this way are positioned in the *eyes band* (on average 8 out of 10) [Fig.5]. This allows us to select the band easily and with high reliability (100% of hits).

Furthermore, we apply the vertical projection to the *eyes band* [Fig.6(a)], finding more than two distinct peaks [Fig.6(b)] due to shadows, hair, and ears. For each distinct peak we take the corresponding grey level piece of image and give it as input to a multilayer perceptron which has been trained to distinguish images represent-

Figure 3. Some example of ”clustered” eyes

Figure 4. Eyes template

ing eyes from those representing other features that can be found in the same band. The neural network has one hidden layer with ten units and two outputs neurons; it has been trained by backpropagation on 200 images and tested on 200 images. The classification error is about 9% but only very rarely the network is wrong on both eyes, so we know that an error has been made since we have a face with a number of eyes different from two. This allows us to treat this situation separately and we are studying a way to correct the error.

Finally, the two sub-images chosen in the previous step are refined in order to tailor them around each eye. For this aim, we calculate the vertical derivatives that highlight the eyebrows: the eyebrow position gives the eye upper and lateral limits. As regard the lower eyes limits, we look for them in the corresponding position of the *feature image* where the iris are very evident.

3.2 Lips and nose localization

Once determined the eyes' bounding boxes, we move to the lips and nose localization using both grey level and color information.

On the grey level image a non-linear edge detector is applied [12]. The detector uses local statistical information: a square window centered around a pixel P is divided into two sub-windows of equal size in four different ways [Fig.7]. For each sub-division i the following function D_i (diversity) is evaluated:

$$D_i = \alpha \Delta m - (1-\alpha)\Delta\sigma$$

where $\Delta m = |m_a - m_b|$, $\Delta\sigma = |\Delta\sigma_a - \Delta\sigma_b|$, m_a , m_b, σ_a , σ_b are respectively the mean and the standard deviation in the regions a and b, and α is a constant ($0 \leq \alpha \leq 1$).

The maximum diversity $D(P) = \max_{i=1..4}\{D_i\}$ is assigned to P. We use this method with a 7x7 window size positioned on a pixel out of 3. The lips and the nose are roughly characterized by pixels with high horizontal and vertical diversity respectively.

To better localize the lips, we combine color information with the output of the edge detector. Since in the Cr color plane image the pixels corresponding to the lips have a very high value [Fig.8], we threshold the Cr

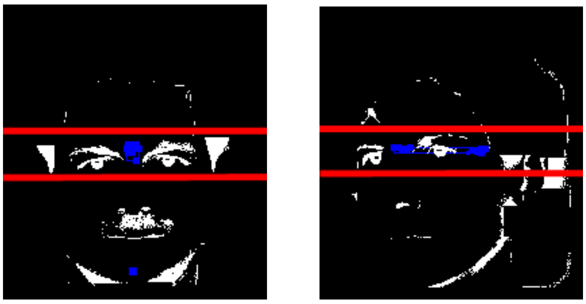
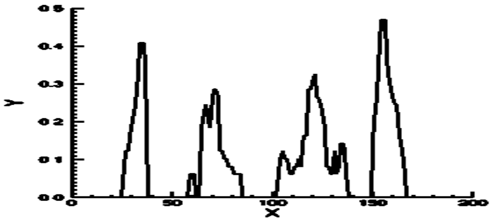


Figure 5. Template matching results and identified eyes band

(a)



(b)

Figure 6. (a) Eyes band; (b) Vertical projection

image in a rectangular region centered around the line identifying the highest number of horizontal edge pixels. The thresholding operation maintains only the 10% of the pixels with the highest values. The bounding box of the largest region found in this way corresponds to the lips sub-image.

The nose is localized in the remaining portion of the image in the region where the horizontal diversity is different from 0.

We have thus identified four sub-images [Fig.9].

4 Identification of fiducial points

In this module we work on the single sub-images separately and at the highest-resolution.

4.1 Eyes

4.1.1 Pupil and eye extreme points recognition

The first step in the recognition of the eye fiducial points is the identification of the pupil.

Some authors (e.g. [13], [14]) suggest finding the darkest pixels. However, this approach is suitable only

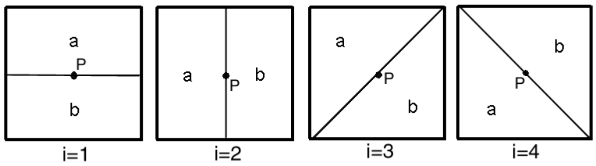


Figure 7. Windows

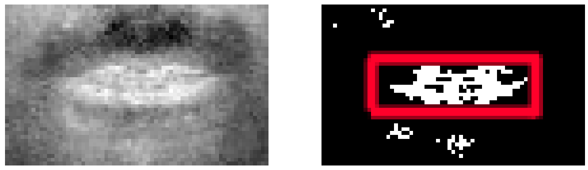


Figure 8. Cr plane and its binarization

to particular illumination conditions. Instead, in general, we cannot guarantee this: as can be seen in figure 10, a reflex spot makes several pixels internal to the pupil very bright.

In our previous work [10] we resorted to template matching. As we looked for the iris, the template had the shape of a circle (with ray equal to 8 pixels). We remind here that at this stage we are working on images of different dimensions which means that the use of a fix template is not suitable. We could introduce adaptive templates, but we found more flexible the use of the Hough transform to look for circumferences. The transform is applied to the binarized horizontal derivative of the eye grey-level image [Fig.11(a)] since it highlights the two iris' vertical sides, which are the best references to find the iris' border. Among the found circumferences we choose the one which obtained the most of the votes [Fig.11(b)]. The centre of the pupil P corresponds to the centre of the found circumference.

Then, the lower extreme of the iris is identified as the intersection of the found circumference and the vertical axis A passing throw P. It is not possible to look for the upper visible extreme of the iris in an analogue way since the point found would exceed the correct position: in most people the iris' upper half is partially occluded by the eyelid.

What we do is to calculate the absolute value of the vertical derivative, $\triangle_y$, of the eye image and to threshold it keeping the 10% of the pixels with the highest values [Fig.11(c)]. The upper extreme of the eye is localized on the axis A in correspondence to the first pixel, starting from the point P, that belongs to the region obtained by the thresholding. This is done after the identification and elimination of reflex spot eventually present in the iris [Fig.10].

4.1.2 Eye corners and upper arc determination

To define the shape of the eye, we apply an edge following algorithm based on the idea of the hysteresis thresh-

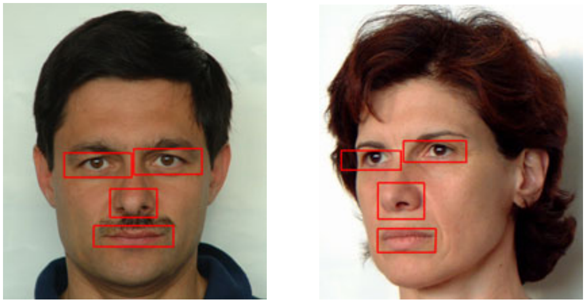


Figure 9. Subdivisions

Figure 10. Eye with a reflex and its localization

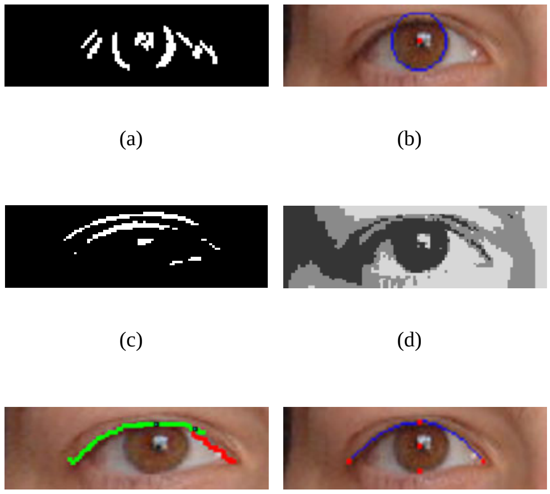
olding [15]. The two thresholds are automatically determined in order to keep the 10% and the 40% of the pixels with the highest values of $\triangle_y$. The contour C obtained in this way [Fig.11(e), *left line*] well-defines part of the shape of the eye, but it does not identify precisely the corners. This is the reason why we have the necessity to adopt another method. We start from the consideration that the white internal part of the eye is quite evident. In order to emphasize the contrast between it and the surrounding regions, we equalize the grey-level image and then cluster it requiring 3 clusters [Fig.11(d)]. The upper border, U, of such white part [Fig.11(e), *right line*] gives a good indication to detect both the internal corner of the eye and its outline: combining U and the contour C, we determine the parabola which better approximates them. Then we localize the eye internal corner on the parabola at the height of the end of U, and the external corner on the parabola at the height of the end of C. This allows us to find a good approximation of the upper part of the eye outline [Fig.11(f)].

4.2 Lips

Our goal for the lips is the determination of the corners and of their middle point at least. To get a more robust estimate, the entire upper lip outline is determined, and, for frontal face images only, also the lower lip outline is described. The technique adopted is based on the following steps: localization of the lips cut and of the lips lowest point, localization of the lips corners and of the lips outline.

4.2.1 Recognition of the lips cut and of the lips lowest point

To determine the lips cut we apply the Sobel vertical derivative operator to the mouth sub-image. We then cluster it into three clusters: one associated to light-to-dark vertical transitions, one to dark-to-light and one to no meaningful transitions [Fig.12(a)]. We then determine



(a) (b)

(c) (d)

(e) (f)

Figure 11. Processing of the eye's image: (a)Input to the Hough transform; (b)Circumference found by the Hough transform; (c)Binarization of the vertical derivative; (d)Clustering of the equalized image; (e)Edge-following; (f)Detection of the eye upper arc and of the eye fiducial points.

the largest dark connected region and we extract, for every x, the upper pixel belonging to it [Fig.12(b)]. The line L connecting these pixels well represents the lips cut apart from its extremes: in some cases, the line ends before the corners, in other, it exceeds them. At this point we are able to recognize the lips lowest point as the lower extreme of the largest connected white region under the line L [Fig.12(b)].

4.2.2 Recognition of the lips corners and of the lips outline

We go back to the original color image considering the mouth box only and cluster it into four clusters. The darkest one identifies the shadow corresponding to the lips cut and gives a precise information regarding the horizontal position of the corners. Combining this with the information given by the line L, we obtain the correct corners position [Fig.12(c)].

We apply the algorithm in the CIE-Luv color space, being the one which has given experimentally the best results consistently, with respect to RGB, CIE-Lab and HSV color spaces.

In order to identify the lips outline, we apply a second time the same clustering algorithm requiring two clusters. This lead to obtain one cluster, M, which is roughly associated to the lips and the other to the surrounding skin. The outline of the region M, is quite precise in its upper part, B, apart from its corners. We thus obtain the upper lips outline straiting B to finish in the found corners.

Moreover, for frontal face images, we can identify the lips lower outline as the parabola passing trough the two corners and the lips lower extreme point [Fig.12(d)].

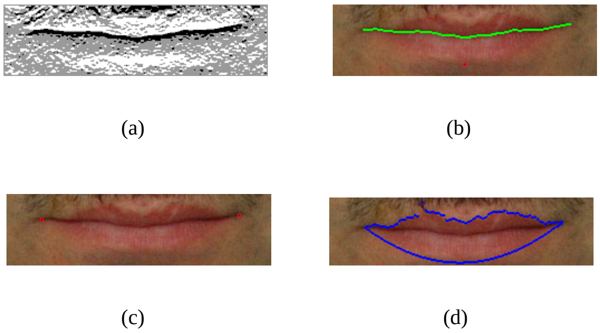


Figure 12. Processing of the lips' image: (a) Clustering of the vertical derivative with 3 clusters; (b) Detection of the lips cut and of the lips lowest point; (c) Corners recognition; (d) Outline.

4.3 Nose

Regarding this feature, we are interested in finding the nose tip. Observing numerous faces' images, we concluded that the nose is characterized by two dark regions, corresponding to the nostrils, and a light region, corresponding to the reflex of the light on the nose tip. In order to identify these regions, we applied the clustering algorithm to the nose grey-level sub-image requiring four clusters. The lightest grey-level pixels correspond to the reflex, and the darkest grey-level pixels correspond to the nostrils [Fig.13]. Moreover we observed that the middle point between the nostrils gives a good vertical localization of the nose tip and that the lower extreme of the region corresponding to the reflex gives a good horizontal localization. The intersection of these two axis gives the nose tip.

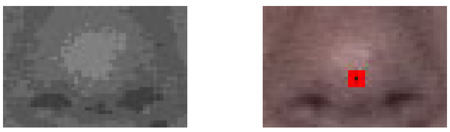


Figure 13. Clustering and Tip of the nose

5 Results and discussion

The method described has been experimented on 200 color images acquired in different illumination conditions. They represent either frontal or rotated faces of Caucasian women and men. We asked to the people to have a neutral expression keeping the mouth closed and the eyes opened. We have not dealt with the case of men with beard.

The algorithm works well on images of very different scales and finds fiducial points using both color and shape information.

The decomposition module localizes the sub-images representing the features of interest with high confidence on both frontal and rotated images. The most of the errors are introduced by the classifications done by the neural network. On our database this module fails on the 5% of the images.

The fiducial points are detected with high accuracy (errors of 1 or 2 pixels are negligible). The only critical point is the eye external corner, where we find an error that, normalized with respect to the eye dimension, can be evaluated as about 10% in the worst case.

Further improvements are required. As regard the sub-images identification, different learning algorithms can be experimented to recognize the eye with higher accuracy (e.g. SVM). As regard fiducial points determination, the algorithm is not completely satisfactory on rotated images, since it does not determine the lips lower outline correctly and it does not localize the nose tip precisely.

A final consideration has to be done about the program running time. The time necessary to process the first module on a Pentium III, 800MHz, 256Mb of RAM is about 5 seconds (the second module computational cost depends strictly on the dimension of the images). The running time can be certainly reduced developing the algorithm in a compiled language and optimizing the code.

References

[1] J. Huang, C. Liu, and H. Wechsler. Eye detection and face recognition using evolutionary computation. In Wechsler et al. [16].

[2] J. Ahlberg. A system for face localization and facial feature extraction. Technical Report LiTH-ISY-R-2172, 1999.

[3] F. Pighin, J. Hecker, D. Lischinski, R. Szeliski, and D. H. Salesin. Synthesizing realistic facial expressions from photographs. *Proceedings Siggraph98*, pages 75–84, 1998.

[4] R. Hsu and A.K. Jain. Face modeling for recognition. *Proceedings IEEE International Conference Image Processing*, pages 693–696, 2001.

[5] A.L. Yuille, P.W. Hallinan, and D.S. Cohen. Feature extraction from faces using deformable templates. *International journal of computer vision*, 8(2):99–111, 1992.

[6] R. Brunelli and T. Poggio. Face recognition: Features versus templates. *IEEE Transactions PAMI*, 15(10):1042–1062, 1993.

[7] R. Herpers and G. Sommer. An attentive processing strategy for the analysis of facial features. In Wechsler et al. [16], pages 457–468.

[8] M. Pardàs and M. Losada. Facial parameter extraction system based on active contours. *Proceedings ICIP, Thessaloniki*, pages 1058–1061, 2001.

[9] T. Kawaguchi, D. Hidaka, and M. Rizon. Detection of eyes from human faces by hough transform and separability filter. *Proceedings ICIP, Vancouver*, I:49–52, 2000.

[10] R. Lanzarotti, N.A. Borghese, and P. Campadelli. Automatic features detection for overlapping face images on their 3D range models. *Proceedings ICIAP, Palermo*, 2001.

[11] M.A. Arbib and T. Uchiyama. Color image segmentation using competitive learning. *IEEE Transactions on pattern analysis and machine intelligence*, 16:1197–1206, 1994.

[12] P. Zamperoni. Feature extraction. In H. Maitre and J. Zim-Justin, editors, *Progress in feature processing*. North Holland, 1996.

[13] R. Stiefelhagen, J. Yang, and A. Weibel. Tracking eyes and monitoring eye gaze. *Proceedings of the Workshop on Perceptual User Interfaces (PUI)*, pages 98–100, 1997.

[14] Y. Yan and K. Challapali. A system for the automatic extraction of 3D facial feature points for face model calibration. *Proceedings ICIP*, II:223–226, 2000.

[15] E. Trucco and A. Verri. *Introductory techniques for 3D computer vision*. Prentice-Hall, Inc., New Jersey, 1998.

[16] H. Wechsler, P. J. Phillips, V. Bruce, F. Fogelman Soulié, and T. S. Huang, editors. *Face recognition*. Springer, London, 1998.