A COLOR-BASED METHOD FOR FACE DETECTION

Paola Campadelli, Francesco Cusmai, Raffaella Lanzarotti

Dipartimento di Scienze dell'Informazione
Università degli Studi di Milano
Via Comelico, 39/41 20135 Milano, Italy
{campadelli, lanzarotti}@dsi.unimi.it
fc501934@silab.dsi.unimi.it

ABSTRACT

We describe a face detection algorithm, which characterizes and localizes skin regions and eyes in 2D images using color information. The method is scale-independent, works on images of either frontal, rotated or tilted faces, with a single person or group of people, and does not require any manual setting or operator intervention. The algorithm has been used successfully as first step of a person identification system.

1. INTRODUCTION

Due to the high inter-personal variability (e.g. gender and race), the intra-personal changes (e.g. pose and expression), and the acquisition conditions (e.g. lighting and image resolution) face detection is not a trivial task and it is still an open problem.

In the last decade, several works have been presented [14]; more recently color has been considered a useful information, since it supplies powerful methods in terms of scale and pose independence, as well as robustness to occlusions; the critical point regards the management of very different illumination conditions. In [9], [11], [12], [13] the skin color distribution is modelled with a Gaussian mixture. Such approach seems reasonable in consideration of the fact that human skin forms a relatively tight cluster in color space, even when different races are considered [8]. Another advantage is that, as shown in [13], a simple mixture of two bidimensional Gaussians is suitable for modeling the skin color distribution. The main problem of this method regards the determination of a threshold on the probability measure to decide which pixels correspond to skin: both a fix and an adaptive threshold fail especially in complex scenes and in particularly unhomogeneous illumination conditions.

Recently in [7] a more robust solution has been presented: before segmenting an image, a non linear transformation to the $YPbPr$ color space is applied to make the skin cluster luma-independent.While such approach works well in most of the situations, it still fails in the challenging ones, where, in general, it produces a high percentage of false positive. Such wrong segmentation requires to deal with more uncertainty in the next steps, and in some case it is the cause of errors.

In this paper we propose a method consisting in two steps: at first skin regions are looked for, then an eye detector searches for the presence of eyes within them (validation step).

The skin region detector is based on a Gaussian mixture model and a region growing algorithm that, starting from the pixels with the highest probability of being skin, dilates the corresponding regions taking also into account the presence of borders, which function is to brake the dilation.

The validation step consists in selecting the regions with a typical eye chromatic characterization, and classifying them as eye or not eye by means of Support Vector Machines: if at least one eye is present within the skin region, we conclude that it is a face, otherwise we discard it.

The method is able to detect a face either in the foreground or in groups, independently of scale, head rotation and partial occlusions, providing that at least one eye is visible. It works with very different illumination and background conditions.

The paper is organized as follows: in section 2 the skin region localization algorithm is described; in sections 3 and 4 the eye characterization and detection are presented; in section 5 results and further developments are discussed.

2. SKIN REGION LOCALIZATION

To characterize the skin colors, we model a significant sample S with a Gaussian mixture. According to [13], we adopt a simple and statistically well justified model made of two bidimensional Gaussians each one parameterized by (μ_i, $\sigma_i^2 I$). The parameters are estimated using the EM algorithm [10]. The means μ_i are initialized calculating the clustering [4] of S into two clusters and taking the two centroids, while the variances, σ_i^2, are initialized considering the dispersion of the samples in S.

Regarding the construction of S, we have payed attention to gather samples of different people whose pictures have been taken in very different illumination conditions. In particular, we have collected four million samples taking pictures of 8 caucasian people under 10 illumination conditions.

Using the sample chrominance components, we have built the Gaussian mixture models in the YPbPr color space, having verified its appropriateness as reported in [11].

Once the skin color model is built, in order to obtain a robust skin map, we identify the pixels in the image which have a high probability of being skin and iteratively grow the corresponding regions, making the expansion stop at the face borders. Of course we do not know where the face borders are, thus we look for the most significant edge pixels in the image and modify their colors so that they will have a very low probability of being skin, this is obtained emphasizing the green components.

At each iteration the colors of the found skin map pixels are changed into the color with the highest probability and the image is then low pass filtered. This process has the effect of 'absorbing' the weakest edges, usually due to shadows, while stopping the skin map expansion at the strongest borders. More precisely, we designed the following algorithm:

- Input: I= Original RGB Color Image
- Initialization:
  1. Transform the RGB image into the $YPbPr$ color image
  2. Calculate the pyramids of the three planes Y, Pb, Pr
  3. For each level l of the pyramids, and for each color plane, extract the edge pixels by means of Sobel edge detector, and threshold them
  4. Construct the border image B: each edge image votes for all its edge pixels, and the votes are added in B, which is finally normalized.
  5. Build I^0: $(I^0) = (Red(I), Green(I) + B \cdot (255 - Green(I)), Blue(I))$
- Iteration for $t = 0, ..9$:
  1. if $t > 0$:
     - $I^t = I^{t-1}$
     - $\forall$ *pixel* $p \in S^{t-1}$, $I^t(p) =$ the color in the image with the highest probability to be a skin color pixel
     - $I^t = Gauss(I^t)$, where *Gauss* is a Gaussian filter (5×5) applied to each color plane
  2. Calculate the probability image P^t, such that for every $p \in I^t$, $P^t(p)$ is the probability of being a skin color pixel
  3. Determine the thresholds $th_1 = [8/10 \cdot max(P^t)]$ and $th_2 = [1/10 \cdot max(P^t)]$
  4. Calculate the skin map S^t thresholding P^t with th_1, and aggregating to the obtained regions the pixels p such that $P^t(p) > th_2$
- Final skin map: to each connected region in S^9, apply a $n \times n$ morphological closing operator with n proportional to the region area.
- Output: The final skin map

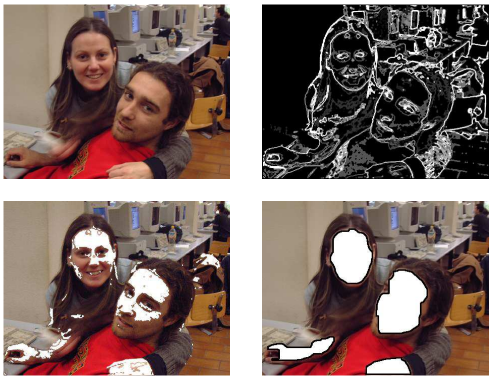

Fig. 1. Face localization: Original image; B: Border image; Initial skin map; Final skin map

In figure 1 we show the output of the most significant steps of the algorithm on an image with two people in foregrounds. It can be seen that all the skin regions are localized very precisely and that to discriminate between faces and non faces a validation criterion is necessary (see next paragraph).

A comparison of our skin localization algorithm with the one presented in [7] can hardly be made since we do not know which images of the cited database [1] have been used by the authors for their test. To have an idea of the different performances, we made an implementation from their description. Our algorithm, although computationally more expensive, is much more selective in the skin identification (see fig. 2) making the subsequent validation step easier.

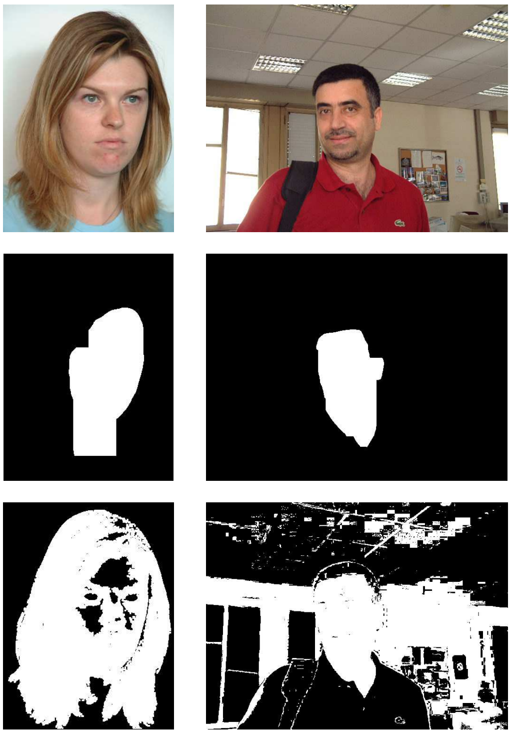

Fig. 2. Some result: *First line*: original images; *Second line*: Our skin maps; *Third line*: Skin maps obtained with our implementation of the algorithm described in [4]

3. EYE CHARACTERIZATION

To determine if a skin region corresponds to a face or not, different criteria could be adopted [7], [14]. The method we propose verifies whether at least one eye is present in the skin region; to this end, rather than examine the whole skin region, we first restrict the research area localizing the potential eyes, and then we validate them by means of a classifier.

As observed in [7], the eyes are characterized in the *PrPb* planes by a low red component and a high blue one; on the basis of that, the authors defined the following *EyeMap* transformation:

$$EyeMap = \frac{1}{3}\{(P_b^2) + (\hat{P}_r)^2 + (P_b/P_r)\}$$

where P_b^2, $(\hat{P}_r)^2$, and P_b/P_r all are normalized to the range $[0, 255]$ and $\hat{P}_r$ is the negative of P_r (i.e $255 - P_r$).

Since it can happen that the *EyeMap* has high values in correspondence to the mouth too, we calculate also the *MouthMap* with the transformation:

$$MouthMap = (255 - (P_r - P_b)) \cdot P_r^2$$

we binarize it maintaining the 4% highest values, and put to 0 in the *EyeMap* the pixels corresponding to the mouth. At this stage the *EyeMap* image is thresholded, maintaining the 20% of the highest values.

Moreover, examining the eye luminance values, we observe that they are always among the darkest within the skin region. Thus, to strengthen the eye localization, we threshold the normalized gray level portion of image corresponding to the considered skin region, maintaining the pixels with a gray level lower than 0.3, and we multiply it with the binary *EyeMap*.

To distinguish between eyes and all the other possible regions identified by this module we have trained a Support Vectore Machine (C-SVM) with RBF parametrized by $\gamma = \frac{1}{2\sigma^2} = 0.005$, and $C = 1$ for the *Eye Class* and $C = 3$ for the *Non Eye one* thus penalizing false positives. With this setting we have obtained an error of 2.7%, having used a training and a test set respectively of 4857 and 4819 gray level images taken from the FERET Database [2].

In the following we describe the eye detection method based on the trained support vector machine classifier.

4. EYE DETECTION

Starting from the regions r highlighted in the previous section, we try to localize the eyes, if present.

In order to speed up the procedure, we first extract a sub-set of pixels from each region r: we consider only the positions corresponding to the vertices of a lattice overlapped to r, whose spacing is proportional to the dimension and regularity of r.

More formally, being A_r the area of r, and $A_{BB(r)}$ the area of the bounding box strictly enclosing r, we define the lattice spacing C_r as:

$$C_r = \left\lceil \left(\frac{A_r}{\pi} \cdot \frac{A_r}{A_{BB(r)}} \right) /1.5 \right\rceil$$

For each visited point, we extract 3 candidates subimages which vary in scale, we give them as input to the C-SVM classifier, and we construct a cumulative image C_{img} where to each vertex of the lattice is attributed the sum of the output obtained at the different scales, interpreting them as the distance of the subimage from the margin.

We extract the positions c_i in which the C_{img} values are greater than $th_1 = 0.7 \cdot max(C_{img})$, and we cluster them according to their mutual distance, determining the cluster centroids.

Among the positions in C_{img} which are 'close' to the found centroids, we determine the ones whose values are greater than $th_2 = 0.1 \cdot max(C_{img})$, obtaining the points a_j.

We then calculate the final centroids which take also into account the second group of found points but associating to them a lower weight. Such centroids represent the potential eye positions.

Finally we calculate for each centroid n a vote v_n, according to the following rule:

$$v_n = \frac{t \cdot \sum_{k=1}^{t} C_{img}(c_t) + z \sum_{j=1}^{z} C_{img}(a_j)}{e}$$

where: c_i are the points whose centroid is n and whose values in C_{img} are greater then th_1; t is the cardinality of the set c_i; a_j are the points whose centroid is n and whose values in C_{img} are greater then th_2; z is the cardinality of the set a_j; e is the number of visited pixels in a square window centered in n and with side equal to $8 \cdot C_r + 1$.

A position n is classified as 'eye', if the corresponding v_n is greater than the threshold $th_3 = 0.5$.

In order to determine th_3, we have examined the distributions of the v_n corresponding to eye and not-eye regions extracted from 100 face foreground images. Regarding the eyes, we obtained a mean value equal to 2.16 and a standard deviation equal to 1, while the corresponding values for the not-eyes are 0.58 and 0.65. Moreover we observe that 162 eye regions have arrived to this validation step, against the 26 corresponding to not-eyes. This consideration highlights the fact that most of the not-eye regions have already been discarded in the previous steps, allowing at this stage to adopt a threshold which captures also the queues of the eye distribution.

5. RESULTS AND DISCUSSION

We have experimented the face detection algorithm both on images representing single faces, and on images representing groups of people.

For the test on single face images we have chosen the XM2VTS Database [3], and ours, for a total of 1000 images.

For the test on groups of people we have taken a set of 20 images acquired in our laboratory.

The XM2VTS Database consists of 750 true color images of face foregrounds. The people differ for the race, the expression and the age. The images have been taken with a blue background, good illumination conditions, frontal pose.

Our database consists of 250 true color images of face foregrounds. The people are all Caucasian, and they differ for pose, and age. The images have been taken at different scales with a light and homogeneous background, and good illumination conditions.

In table 1 we report the results obtained on the 1000 foreground images, organized according to the number of eyes correctly found, and the number of false positives.

N. of eyes correctly classified	N. of false positives	% of the total
2	0	63.8%
2	1	1.9%
2	2	0.3%
1	0	21.7%
1	1	3.0%
1	2	0.3%
0	0	6.0%
0	1	2.5%
0	2	0.5%

Table 1. Eye detection results obtained on a database of 1000 foreground color images.

The obtained results are encouraging, considering that with the objective of validating a face we arrive to a success in 91% of the cases, that is when at least one eye is detected.

However, when this module is used as first step of a face recognition system, it is important to detect both and only the two eyes. Without any further consideration, we would have too low performances: 63.8% of hits, while, under the assumption of verticality and exploiting the information about the skin map (dimension, position,...), we can improve significantly the system performance. In particular, in order to solve the ambiguity due to the presence of false positives, we select the two positions which are approximately at the same height and at a plausible distance; regarding the case in which only one eye has been correctly localized, we can localize the other at the same height and at an estimated distance, while discarding the false positives considering their relative positions within the skin map. These considerations allow us to detect two and only two eyes in the 91% of the cases.

Once localized the two eyes, we can easily detect the other facial features (nose, mouth, and chin) and identify for each of them the fiducial points [6]. The face is characterized by a vector, called *jets vector*, obtained applying a bank of Gabor wavelet filters to each fiducial point [5]. In order to recognize a test image, we calculate a similarity measure between its *jets vector* and the ones corresponding to the images in the gallery: the test image will be recognized to be the image in the gallery which maximizes the similarity measure. We have experimented the whole recognition system on 600 images of the XM2VTS Database [3], obtaining 92% of hits.

The test on images representing groups of people has been carried out on a set of 20 images representing 2, 3 or 4 people, for a total of 60 people. The algorithm has detected correctly 55 people, and has given only 3 false-positives. We observe that in this context the skin detection module plays a more important role, since it has to deal with more complex scenes; our algorithm has shown high selectivity, producing very few false positives.

Concluding, the system has shown to be robust to different scales, illumination conditions, head poses, and contexts. We are now working on the eye detection module, experimenting other support vector machines; this would make the system more robust, allowing to have higher performances, even without the assumption of head verticality. Moreover we are interested in experimenting the method on a larger database of groups of people, in order to highlight the potentiality of the system and its eventual problems.

6. REFERENCES

[1] The champion database. *Web address: http://www.libfind.unl.edu/alumni/events/breackfast-for-champions.htm*, 2001.

[2] The feret database. *Web address: http://www.itl.nist.gov/iad/humanid/feret/*, 2001.

[3] The xm2vts database. *Web address: http://www.ee.surrey.ac.uk/Research/VSSP/xm2vtsdb/*, 2001.

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

[5] S. Arca, P. Campadelli, and R. Lanzarotti. A face recognition system based on local feature analysis. *Accepted for the 'International Conference on Audio- and Video-based Biometric Person Authentication', to be published in Lecture Notes in Computer Science - AVBPA2003*.

[6] P. Campadelli, R. Lanzarotti, and C. Savazzi. A feature-based face recognition system. *Accepted for the 'International Conference on Image Analysis and Processing', to be published in IEEE Proceedings ICIAP2003*.

[7] R. Hsu, M. Abdek-Mottaleb, and A.K. Jain. Face detection in color images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(5):696–706, May 2002.

[8] M. Hunke and A. Waibel. Face locating and tracking for human-computer interaction. *28th Asilomar Conference Signals, Systems and Comuters, California*, 1994.

[9] S. McKenna, S. Gong, and Y. Raja. Face recognition in dynamic scenes. *British Machine Vision Conference (BMVC97), Essex*, 1997.

[10] G.J. McLachlan and T. Krishnan. *The EM algorithm and extensions*. John Wiley & Son, 1996.

[11] J.C. Terrillon, M.N. Shirazi, H. Fukamachi, and S. Akamatsu. Comparative performance of different skin chrominance models and chrominance for the automatic detection of human faces in color images. *Proceedings of the IEEE International conference of Face and Gesture Recognition*, pages 54–61, 2000.

[12] H. Wu, Q. Chen, and M. Yachida. Face detection from color images using a fuzzy pattern matching method. *IEEE Transactions Pattern Analysis and Machine Intelligence*, 21:557–563, 1999.

[13] M. Yang and Narendra Ahuja. Gaussian mixture model for human skin color and its applications in image and video databases. *SPIE Proceedings Storage and Retrieval for Image and Video Databases VII, 01/23 - 01/29/1999, San Jose, CA, USA*, pages 458–466, 1999.

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