Dipartimento di Informatica e Automazione
Via della Vasca Navale, 79 – 00146 Roma, Italy
http:\\www.dia.uniroma3.it

Face detection with colour segmentation and fuzzy template matching

Marco Boccioli[1], Stefano Panzieri[2],
José Luis Díez Ruano[3]

[1]CERN, Physics Department of the European
Organization for Nuclear Research,
Geneva, Switzerland
marco.boccioli@cern.ch

[2]Dipartimento di Informatica e Automazione,
Università di Roma Tre,
Rome, Italy
panzieri@dia.uniroma3.it

[3]Departamento de Ingeniería de Sistemas y Automática,
Universidad Politécnica de Valencia,
Valencia, Spain
jldiez@isa.upv.es

RT-DIA-112-2006
November 2006

ABSTRACT

On this paper it is shown a study and integration of different procedures in order to detect a face in a still picture or in a movie sequence. Once a skin colour region has been detected by colour segmentation, it is then classified as face or non-face through a template matching method. This method has been improved by the use of fuzzy theory. After face detection, face features, e.g. eyes and mouth, are then extracted.

1 Introduction

Human face detection plays an important role in such applications like video surveillance or human-machine interfaces. Since face can be considered as a window of social life mechanisms, it can be used for translating communication wills into machine operations. Face and eyes tracking belong to this topic.

The aim of this project is to merge some of the methods for image processing in order to create a face detection system.

Face detection can be assumed as a two-class classification problem: face or not-face. Some of the techniques for face detection use template approaches [3] [20], color segmentation methods [10] [16] [12]. Table 1 shows a sample of face detection approaches.

This paper illustrates a technique for detecting the effective presence of a face and tracking the face and its features. Face detection is the first step used for face tracking. In order to select candidates from colour regions in the input image, it is necessary to choose an appropriate colour space. Normalized RGB representation was preferred, as it has the advantage of having modest light conditions dependence [15].

A threshold is used for extracting skin colour-like regions. Through region segmentation, only one face candidate is then selected. Colour-based approaches find difficulties in robust facial detection. Hence, a template matching method was found to be useful for determining face presence. As it is impossible to merely distinguish a pixel simply belonging either face or non-face class, template matching results more accurate if combined with fuzzy theory [8]. Despite of the method used in [8], this project does not use template matching in the whole image, but just inside the face candidate region previously detected. As template matching is performed with a high computational cost, its use inside a more limited region allows process time reduction. Once a face has been detected, it is then possible to localize its features like eyes, mouth or nose.

This system can be divided into two main stages: face detection and feature extraction. Both of these stages make use of colour information and spatial moment descriptors [2]. Face detection is based on template matching and fuzzy theory. Feature extraction integrates colour [9] with edge [3] information. Fig. 1 shows the developed face detection algorithm.

Author	Approach	Used Characteistics	Face Pose
[9] [11] [10] [16]	Segmentation, colour models	Colour, edges	Frontal, semi frontal
[15]	Markov models	Colour	Frontal
[8]	Template matching	Colour	Frontal a lateral
[19] [3]	Template matching	Intensity, edges	Frontal
[2]	Colour models	Colour	Frontal
[6] [7]	Hough transform, Template	Intensity	Frontal, semi frontal
[19]	Learning	Colour	Frontal, lateral
[1]	Neural networks	Intensity	Frontal, lateral

Table 1: *Face detection approaches*

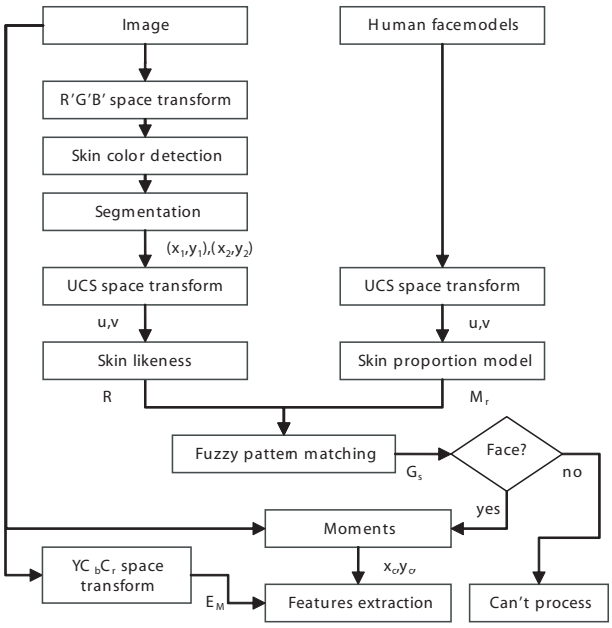

Figure 1: *Face detection algorithm*

2 Face decetcion

2.1 Skin region detection

Except for albinos, human skin colour (hue) is the same, independently from human race [2]. A darker skin has just higher colour saturation. This consideration is useful for detecting skin colour regions. In order to classify colours, it is important to correctly represent them. After a comparison between different colour representation spaces [17] [5], depending to experimental results, is has been chosen a different colour space for each stage of the processing.

RGB space is the most common colour space, but it is strongly luminance dependent. A solution to this problem is to normalize it:

$$C' = C/(R + G + B), \qquad C = R, G, B \tag{1}$$

so that the subspace $R' \times G'$ can be used for finding flesh areas in the image regardless of large lighting variations. The blue component B' has been discarded after having realized that it does not furnish any interesting information about flesh colour.

Fig. 2 shows the difference between skin regions (green) and background regions (red) in R, G and in normalized R, G: the latter is evidently more accurate in separating the two classes. Other colour spaces, such as $YCbCr$ [9] and Uniform Colour System (UCS, through Yuv) [8], have been preferred for feature extraction after having tested results given by each method.

In order to create a decision threshold for *skin, non-skin* classification, it is necessary to collect an amount of empirical skin colour values. This takes advantage of region texture technique [11]. Candidate skin region values are compared with a threshold computed a

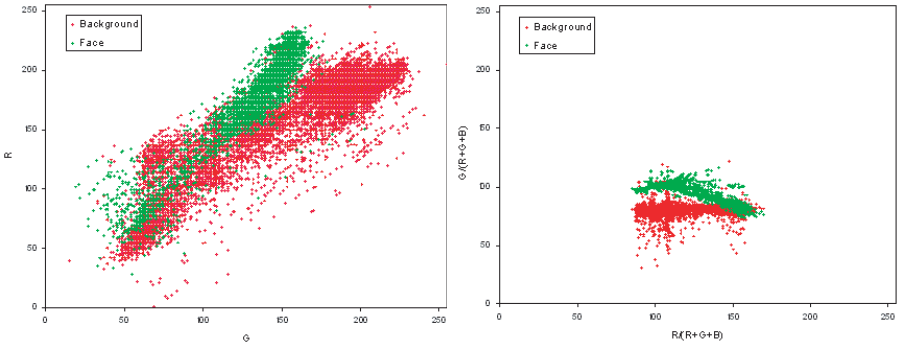

Figure 2: *R, G colour space (left); Normalized R, G colour space (right)*

Figure 3: *Skin region detection*

priori:

$$p(x,y) \in \text{skin} \Leftrightarrow \begin{cases} R'(p(x,y)) & \in \{96,..,120\} \\ \quad \wedge & \\ G'(p(x,y)) & \in \{75,..,86\} \end{cases} \tag{2}$$

where $p(x, y)$ is pixel at (x, y) coordinates, and R', G' are normalized R, G colour components. These thresholds where determined through experiments in order to minimize the proportion between false positives and positives.

Fig. 3 illustrates skin colour region extraction from input image.

The extracted region is then segmented through region growing [17] [10] and labelling [13], so that a more accurate face candidate region can be obtained. The effects of these methods are shown in Fig. 4.

Based on the hypothesis of one single face present in the picture, only the biggest candidate region is selected. This procedure is not affected by other body parts, i.e., arms, because their surface is usually minor than head area.

Figure 4: *Region growing (left) and labelling (right)*

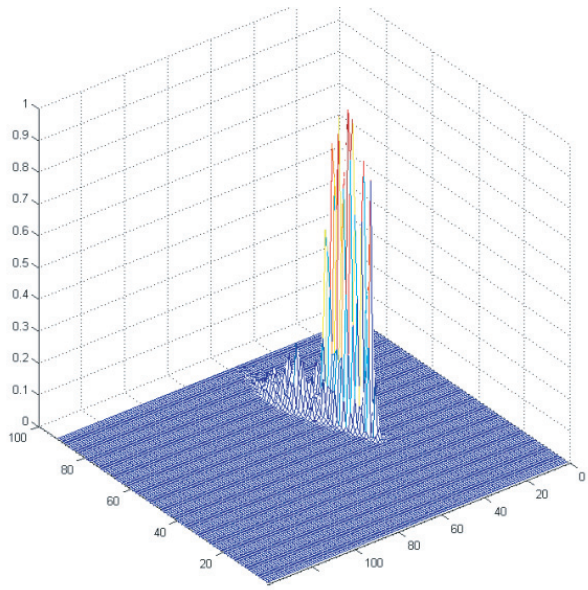

Figure 5: *Fuzzy skin colour distribution model*

2.2 Fuzzy template matching

By applying the pattern matching method [20] [19] [3], face candidate region is classified as *face* or *non-face* through a comparison between candidate region and predefined face models. It is impossible to match two samples simply comparing them pixel by pixel, as, especially along the edges, pixels do not strictly belong just to *skin* and *non-skin* clusters. Hence, this comparison makes use of a fuzzy approach [8]. In order to reach a way of decision more similar to the human one, image is transformed in Uniform Colour System (UCS) space, which is closely similar to human colour perception.

Patterns are created through the following procedure. From each sample picture, skin colour regions are manually extracted. Chromatic components u, v are then extracted to build the fuzzy colour distribution model $I(u, v)$: for each pixel, a skin colour likeness value a_s is collected:

$$a_s\left(p\left(x, y\right)\right) = I\left(u\left(p\right), v\left(p\right)\right) \tag{3}$$

Face templates are then created by collecting different faces in various postures. Five different poses have been selected (e.g.: frontal, lateral right, lateral left, frontal left-rolled, frontal right-rolled; these last two ones have been chosen with and without neck). Each model is a matrix, each cell containing the average skin colour. Their small dimensions reduce diversity details between different persons (Figure 6). The comparison between candidate region and models makes use of fuzzy theory. The region to be classified is divided into as many areas as the number of pattern's cells.

A fuzzy membership function μ_A was defined:

$$\mu_A : R \rightarrow [0, 1] \tag{4}$$

where R describes the likeness between the average skin colour similarity a_s:

$$\bar{a}_s = \frac{1}{n^2} \sum_{p \in F} \rho\left(p\right) \tag{5}$$

Figure 6: *Templates and one of the samples used for templates creation (top-left)*

and the proportion of the skin colour part in a square region of the input image. In fuzzy theory, an type standard function is defined for representing the membership function μ_A:

$$S(x;a,b) = \begin{cases} 0 & a \geq x \\ \frac{2(x-a)^2}{(b-a)^2} & a < x \leq \frac{a+b}{2} \\ 1 - \frac{2(x-b)^2}{(b-a)^2} & \frac{a+b}{2} < x \leq b \\ 1 & b < x \end{cases} \tag{6}$$

where $0 \leq a \leq 1$, $0 \leq b \leq 1$ and $a \leq b$. Parameters a and b control the shape of S. If $a \to b$, then S will behave like a step function. Once $(a,b) = (0.0, 0.6)$ has been chosen, skin proportion is then given by

$$R = \mu_A(a_s) = S(a_s; 0.0, 0.6) \tag{7}$$

The degree of membership is determined by the fuzzy two terms relation

$$A_E(R, M_p) = e^{-a|R-M_r|^b} \tag{8}$$

If the degree of membership between the model and the region reaches a threshold, then the processed region is classified as face.

3 Face tracking

If a face was detected, it is then possible to determine its geometrical and spatial properties like position and pose. For this reason, mathematical methods, i.e. spatial moments, are used [18] [10]. Object centroid can be calculated by the use of p,q-order spatial moment $m_{p,q}$:

$$x_c = \frac{m_{1,0}}{m_{0,0}}, \; y_c = \frac{m_{0,1}}{m_{0,0}} \tag{9}$$

as well as head roll angle θ :

$$\theta = \frac{1}{2}\tan^{-1}\left(\frac{2\left(\frac{m_{1,1}}{m_{0,0}} - x_c y_c\right)}{\left(\frac{m_{2,0}}{m_{0,0}} - x_c^2\right) - \left(\frac{m_{0,2}}{m_{0,0}} - y_c^2\right)}\right) \tag{10}$$

Eigenvalues λ_1, λ_2 represent distribution major and minor axis. They can be computed as follows:

$$\lambda_1 = \sqrt{\frac{a + c + \sqrt{b^2 + (a-c)^2}}{2}} \tag{11}$$

Figure 7: *Face tracking*

$$\lambda_2 = \sqrt{\frac{a + c - \sqrt{b^2 + (a - c)^2}}{2}} \tag{12}$$

where:

$$a = \frac{m_{2,0}}{m_{0,0}} - x_c^2, b = 2\left(\frac{m_{1,1}}{m_{0,0}} - x_c y_c\right), c = \frac{m_{0,2}}{m_{0,0}} - y_c^2 \tag{13}$$

Face window is fit by iterative centroid computation, until convergence [2]. The algorithm is as follows:

1. Compute current centroid inside window.
2. From equations (11) and (12), compute new window dimensions (they may increase or decrease with respect of current dimension).
3. Compute centroid inside new window.
4. If difference between old and new centroid coordinates is inferior than threshold, or if window is shifting outside picture edges, then stop. Else, compute new window dimensions and repeat from the first step.

As result from face tracking algorithm, face centre coordinates and mayor length and width are determined (Fig.7).

4 Features extraction

The next step is to locate face features, i.e., mouth and eyes. A first approach is to detect areas where features are expected to be present. Starting from face image $F(x, y)$, edges are extracted through convolution with Sobel kernel [17] [5]:

$$\frac{1}{4}\begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{bmatrix} \qquad \frac{1}{4}\begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix} \tag{14}$$

Sobel was preferred as it is directional (the first matrix is row kernel S_x, the second one is column kernel S_y), so it permits more accurate search for boundaries along horizontal and vertical axis. This filtering outputs are two images, $F_x(x, y)$ and $F_y(x, y)$, describing borders' horizontal and vertical gradients.

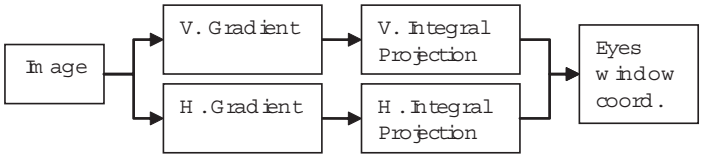

Figure 8: *Eyes window algorithm*

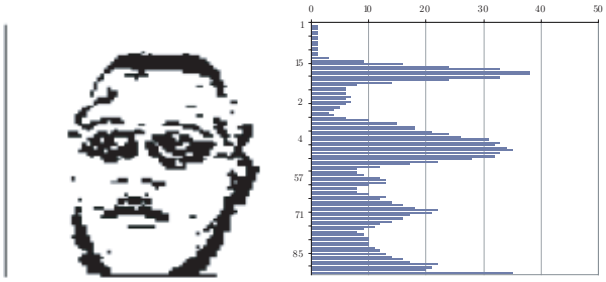

Figure 9: *Vertical gradient*

Once the image being transformed into binary space, integral projection is computed from each of them [3] as shown in Fig. 8. This returns vertical and horizontal occurrences histogram inside the ranges $x_1, \ldots, x_2$ and $y_1, \ldots, y_2$:

$$V(x) = \sum_{x=x_1}^{x_2} F'_x(x,y), \quad H(y) = \sum_{y=y_1}^{y_2} F'_y(x,y) \tag{15}$$

where F'_x and F'_y are binary images obtained by the result of the convolution between input image F and Sobel kernels: $F \otimes S_x$, $F \otimes S_y$.

The analysis of the histograms $V(x)$ and $H(y)$ allows to roughly but quickly locate face features. Eyes y_e coordinate, for example, can be detected by noticing the histogram peak around its 2/4 segment.

Fig. 9 proves how features along vertical gradient are determined.

If face axis inclination passes 9°, integral projection does not return correct histogram. It is then necessary to rotate the image through coordinates changing matrix:

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta & -x_c\cos\theta + y_c\sin\theta + x_c \\ \sin\theta & \cos\theta & -x_c\sin\theta - y_c\cos\theta + y_c \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

in order to get the image normal to the X axis, as Fig. 10 shows.

This method supplies eyes, mouth and nose windows, in which to search for exact features' position.

It is hence possible to locate eye centre coordinates with the following method. $YCbCr$ colour space is used in order to enhance eyes colour features. An eye map E_M is then built under the equation

$$E_M = \frac{1}{3}\left(C_b^2 + \bar{C}_r^2 + \frac{C_b}{C_r}\right) \tag{16}$$

Figure 10: *Face inclination*

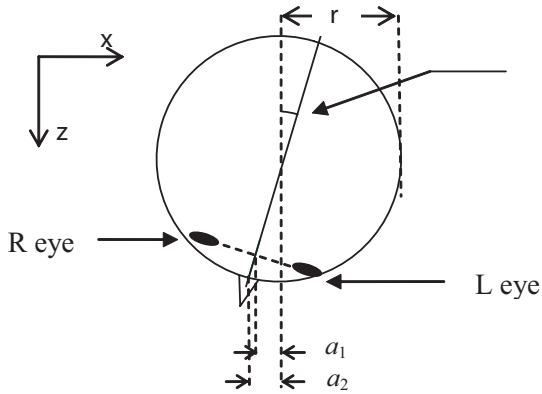

Figure 11: *Head pose*

where colour components C_b^2, C_r^2 and C_b/C_r are normalized to $\{0, ..., 255\}$, and $\bar{C}_r = 255 - C_r$. This equation was built under the observation that regions' colour components surrounding eyes have high blue values (Cb) and low red values (Cr) [9]. $E_M(x, y)$ greyscale map is then used for eye centroid search: inside each eye window, an iterative 1st and 0th order spatial moment is computed. Nose symmetry helps to localize its centre just by computing nose window centre.

Head slope ϕ around y axis (head pose) is computed:

$$\phi = \sin^{-1}\left(\frac{a}{r}\right) \tag{17}$$

where a is the distance between face centroid and face symmetry axis and r is head radius. Looking at Figure 11, at first it was believed to let $a = a_1$, a_1 being the distance between symmetry axis and the projection of the mid-point of the eyes [4]. Nevertheless, it was thought to use $a = a_2$, a_2 being the distance between symmetry axis and nose, as it better approximates the face roundness. Noting that face centroid coordinates (x_c, y_c) are usually different from face symmetry coordinates (x_s, y_s), face simmetry coordinates depend on head angle around z axis (head roll). This does not allow direct use of symmetry coordinates. Hence, (x'_s, y'_s) is used, after changing coordinates (rotation θ) from (x_s, y_s).

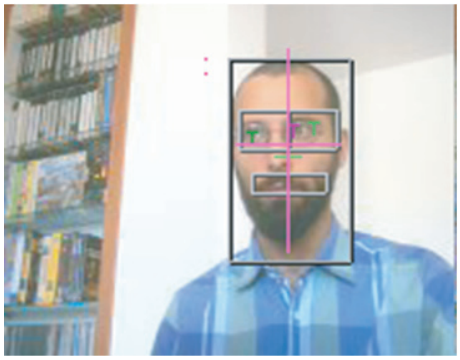

Figure 12: *Final result*

Resolution	Face detection	False positive	Eyes detection	Process Time
160×120	94.1%	15.4%	52.9%	0.38s
176×144	90.3%	11.1%	68.8%	0.55s

Table 2: *Performance summary*

5 Results

The system, implemented in C++, was tested in a computer with 256MB of memory and Intel Pentium 3 processor. Results are obtained from processing over 2000 pictures, with different faces in different positions. Still pictures were taken from a face database available in Internet [14]. For video sequences, a common webcam has been used. Processed pictures and video sequences resolutions were both 160×120 and 176×144 pixels. Results are summarized in Table 2, where the first percentage deals with face detection success after processing images containing one face which dimensions are between 120×120 and 20×20 pixels. More details about this measure can be appreciated from Figure 13, each bar collecting results from a range of different face sizes.

Face detection success in detection is almost independent on face roll θ. Measures were collected from images of constant distance between ace and camera, with approximately 80×80 (± 10) face size. Each element clusters a set of 20 degrees ($\pm 20°$). From this histogram it is evident that face detection reaches a high percentage of success for face window size bigger than 40 pixels.

False positives arrive at 11.1% for higher resolution, and reaches 15.4% for lower resolution. This occurs mainly when wooden objects appear in an image. Of course, the percentage of face detection and false positives depends on the thresholds set for template matching and for skin detection algorithms.

Graph in Figure 14 shows face roll measured and actual angle. Actual angle was manually obtained with the aid of a ruler positioned at the top of the person and a photo editing program. In the samples used for this measure, face size was 90 ± 10 pixels, and

Face size	This project	[8]	[15]	[9]
bigger than 100×120	98%	100%	96%	91%-99%*
bigger than 50×60	99%	97%		
bigger than 20×24	73%	100%		

Table 3: *Detection rates: results comparisons (*depending on detection method)*

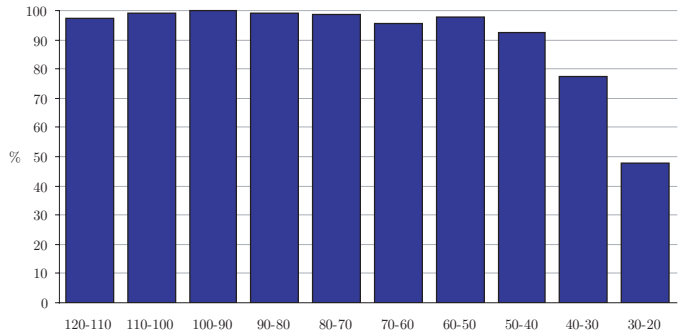

Figure 13: *Face detection success depending on face size; image:* 176×144 *p.*

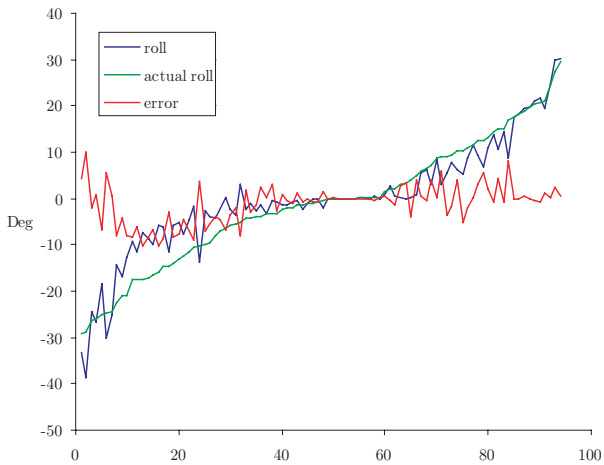

Figure 14: *Face roll measure*

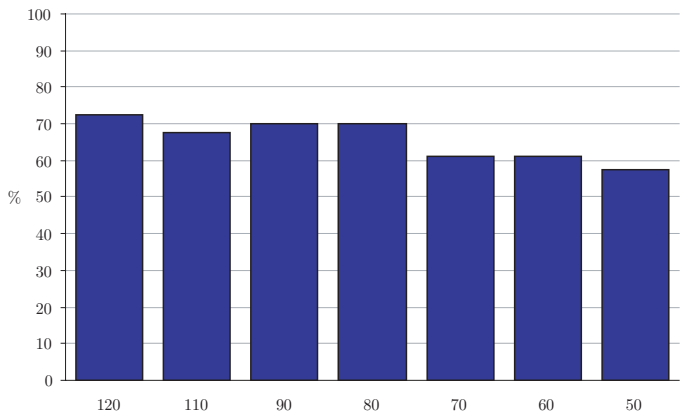

Figure 15: *Eyes detection success in different face sizes; image size:* $176 \times 144p$

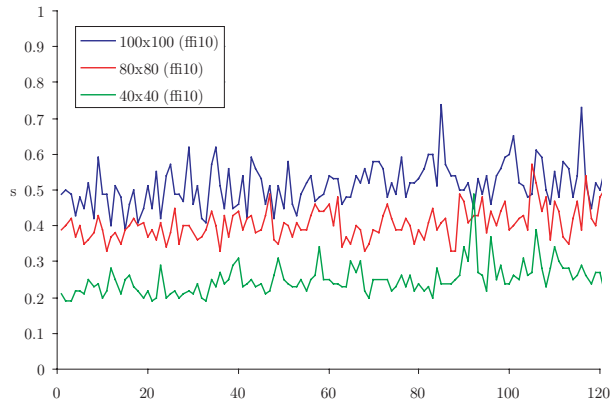

Figure 16: *Process time*

head roll range was $\theta \in [-30°, 30°]$. Eyes detection is described in histogram of Figure 15, where samples where classified depending on face size. In this case, detection ratio depends on face dimension.

Process time depends not only on image resolution but also on background complexity and face size. As shown in Figure 16, time approximates 0.4 seconds for 80×80 sized faces in images of 160×120 pixels. For 176×144 sized images, process time average is 0.1s higher.

Table 3 shows how this project's detection performance is comparable to the results from other proposed methods. Detection rate in the case of faces bigger than 20×24 is lower than in other cases. This is due to the limited face candidate size, which often brings to complexity in deciding whether a blob geometric features belong to face characteristics. It is important to underline that a great difficulty was found in comparing other projects detection performances, as each one uses a different measure and different hypothesis: for instance, looking at performance given by [9] and [15], face size range was not reported. Hence, it is not possible to compare this method's 73% rate with the percentage reported by [15], as face size hypotheses are not the same.

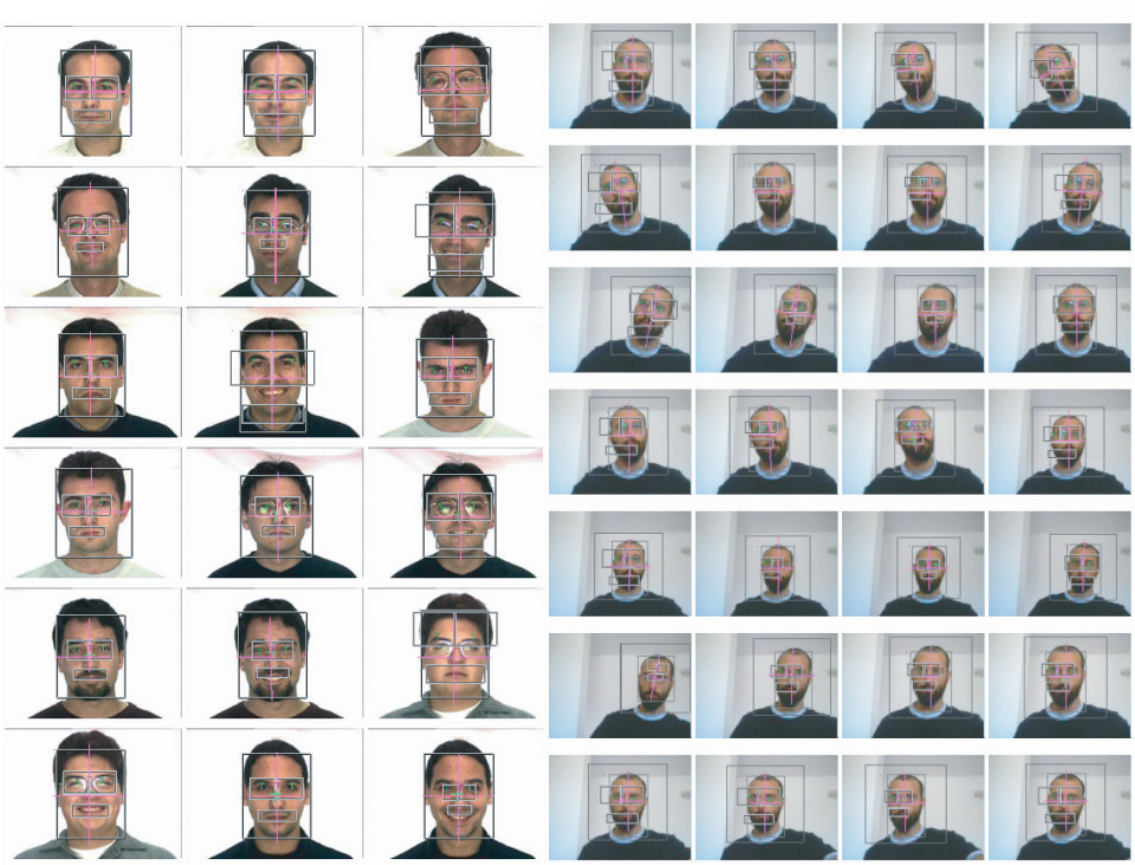

Figure 17: *Face detection. Left: different faces; right: video sequence*

6 Conclusions

This project makes use of some of the techniques applied to image processing and face detection. Different approaches were compared and chosen. Once face candidate region has been detected through colour region extraction, face presence is then proved by fuzzy template matching and it is processed for face tracking and features extraction. This can be a starting point for face recognition or face controlled appliances.

REFERENCES

1. Agui T., Kokubo Y., Nagashashi H., and Nagao T. Extraction of face recognition from monochromatic photographs using neural networks. In *Proc. Second Int. Conf. on Automation, Robotics, and Computer Vision,*, volume vol. 1, pages pp. 18.8.1–18.8.5, 1992.
2. Bradski G.R. Computer vision face tracking for use in a perceptual user interface. *Intel Technology Journal Q2*, 1998.
3. Brunelli R. and Poggio T. Face recognition: features versus templates. *IEEE transactions on pattern analysis and machine intelligence*, vol. 15(no. 10):pp. 1042–1052, 1993.
4. Fu Jie Huang and Tsuhan Chen. Tracking of multiple faces for human-computer interfaces and virtual environments. *Electrical and Computer Engineering.*
5. Gonzales R.C. and Woods R.E. *Digital image processing.* Addison Wesley 2nd ed., 2003.

6. Govindaraju V., Sher D.B., Srihari R.K., and Srihari S.N. Locating human faces in newspaper photographs. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, pages pp. 549–554, San Diego, CA, 1989.

7. Govindaraju V., Srihari S.N., and Sher D.B. A computational model for face location. In *Proc. Third IEEE Int. Conf. on Computer Vision*, pages pp. 718–721, Osaka, JP, 1990.

8. Haiyuan Wu, Quian Chen, and Masahiko Yachida. Face detection from color images using a fuzzy pattern matching method. *IEEE transactions on pattern analysis and machine intelligence*, vol. 21(no. 6), 1999.

9. Hsu R.L., Addel-Mottaleb M., and Jain A.K. Face detection in color images. *IEEE transactions on pattern analysis and machine intelligence*, vol. 24(no. 5):pp. 696–706, 2002.

10. Jianping Fan and Yau D.K.Y. andElmagarmid A.K. andAref W.G. Automatic image segmentation by integrating color-edge extraction and seeded region growing. *IEEE transactions on image processing*, vol. 10(no. 10):1454–1466, 2001.

11. Junchul Chung, Kyongpil Min, Korack Park, and Jaegi Son. An automatic extraction of face and facial feature from face images using skin color and active contour model. *IEEE transactions on pattern analysis and machine intelligence*.

12. Kumar R.T., Raja S.K., and Ramakrishnan A.G. Eye detection using color cues and projection functions. volume vol. 3, pages pp. 337–340, 24-28 June 2002.

13. Lanitis A., aylor T.F., and Cootes T.F. An automatic face identification system using flexible appearance models. *Image and Vision Computing*, vol. 13(no. 5):pp. 393–401, 1995.

14. Martinez A.M. and Benavente R. The ar face database. Technical Report no. 24, CVC, 1998.

15. Oliver N., Pentland A., and Lafter F.B. A real-time face and lips tracker with facial expression recognition. *Pattern Recognition Society 33*, pages pp. 1369–1382, 2000.

16. Phung S.L., Bouzerdoum A., and Chai D. A novel skin color model in ycbcr color space and its application to human face detection. In *Proc. of Int. Conf. on Image Processing*, volume vol. 1, pages pp. 289–292, 22-25 Sept. 2002.

17. Pitas I. *Digital Image processing Algorithms*. Prentice Hall, 1993.

18. Pratt W.K. *Digital Image processing*. John Wiley & sons., 1991.

19. Seo K., im W., Oli C., and Lee J. Face detection and facial feature extraction using colour snake. *IEEE transactions on pattern analysis and machine intelligence*, 2002.

20. Yuille A., Hallinan P., and Cohen D. Feature extraction from faces using deformable templates. *Int. J. Computer Vision*, vol. 8(n. 2):pp. 99–111, 1992.