Proceedings of the 17th World Congress
The International Federation of Automatic Control
Seoul, Korea, July 6-11, 2008

Face detection with colour segmentation and fuzzy template matching

Marco Boccioli*, Stefano Panzieri, José Luis Diez *****

** Physics Department, European Organization for Nuclear Research, Geneva, Switzerland (e-mail: marco.boccioli@cern.ch)*
*** Dipartimento di Informatica e Automazione, Universitá di Roma Tre, Rome, Italy, (e-mail: panzieri@dia.uniroma3.it)*
**** Departamento de Ingeniería de Sistemas y Automática, Universidad Politécnica de Valencia, Valencia, Spain (e-mail:jldiez@isa.upv.es)*

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 track it 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 to translate 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 an improved 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 (Brunelli, 1993, Yuille, 1992), colour segmentation methods (Fan, 2001, Phung, 2001, Thilak Kumar, 2001).

This paper illustrates a technique for detecting the presence of a face and for 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 (Oliver, 2000).

A threshold is used to extract skin colour-like regions. Through region segmentation, only one face candidate is then selected. Colour-based approaches find difficulties in robust facial detection. Therefore, a template matching method is useful for determining face presence. As it is impossible to simply distinguish a pixel simply belonging either to face or non-face class, template matching results more accurate if combined with fuzzy theory (Wu, 1999). Despite of the method used in (Wu, 1999), 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 processing time reduction. Once a face has been detected, it is then possible to localize its features like eyes, mouth or nose.

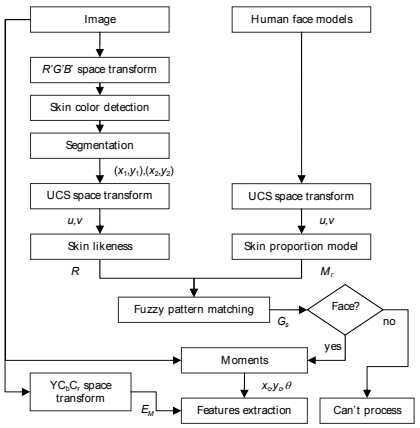

Fig. 1. Face detection algorithm

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 (Bradsky, 1998). Face detection is based on template matching and fuzzy theory. Feature extraction integrates colour (Hsu, 2003) with edge (Brunelli, 1993) information. Fig. 1 shows the developed face detection algorithm.

The paper presents a proposal of techniques to be used for face detection (section 2), face tracking (section 3), and feature extraction (section 4) where fuzzy template matching is included in the face detection algorithm. The results of this approach are presented in section 5.

2. FACE DETECTION

2.1 Skin region detection

Except for albinos, human skin colour (hue) is the same, independently from human race (Bradsky, 1998). 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 (Pitas, 1993, Gonzalez, 2002), 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 its problem is normalization:

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

The subspace $R' \cdot G'$ can be therefore used to find 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.

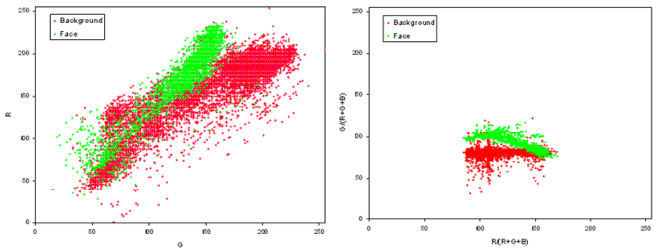

Fig. 2. R;G colour space (left); Normalized R;G colour space (right)

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* (Hsu, 2003) and Uniform Colour System (UCS, through *Yuv*) (Wu, 1999), 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 the region texture technique (Chun, 2004). Candidate skin region values are compared with a threshold computed a priori:

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

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 (Pitas, 1993, Fan, 2001) and labelling (Lanitis, 1995), 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 the head area.

Fig. 3. Skin region detection

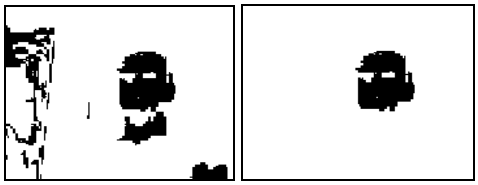

Fig. 4. Region growing (left) and labelling (right)

2.2 Fuzzy template matching

By applying the pattern matching method (Yuille, 1992, Seo, 2002, Brunelli, 1993), 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 (Wu, 1999). In order to reach a way of decision more similar to the human one, image is transformed in 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 as is collected:

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

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 (Fig. 6).

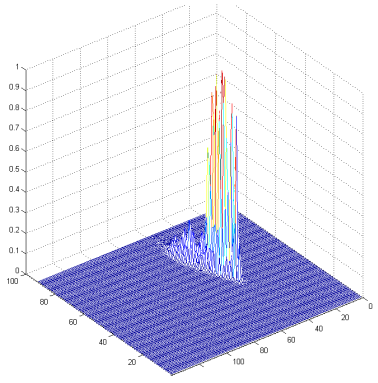

Fig. 5. Fuzzy skin colour distribution model

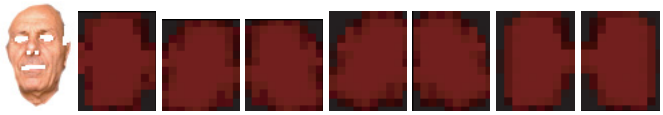

Fig. 6. Templates and one of the samples used for templates creation (top left)

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 patterns 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:

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

and the proportion of the skin colour part in a square region of the input image. In (5), ρ is the likeness function of the pixel p, F is the whole pixel region, n is the total number of cells in the region. In fuzzy theory, a standard function is defined for representing the membership function μ_A:

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

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 $a \rightarrow b$, 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

$$A_E(x_1, x_2) = e^{-a|x_1 - x_2|^b} \tag{8}$$

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 (Pratt, 1991, Fan, 2001). Object centroid can be calculated by the use of p,q-order spatial moment $m_{p,q}$:

$$x_c = m_{1,0}/m_{0,0}, \quad y_c = m_{0,1}/m_{0,0}, \tag{9}$$

as well as head roll angle θ:

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

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

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

$$\lambda_2 = \sqrt{\left(a + c - \sqrt{b^2 + (a-c)^2}\right)/2}, \tag{12}$$

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 (Bradsky, 1998). The algorithm is as follows:

1. Compute the current centroid inside window.

2. From (11) and (12), compute new window dimensions (they may increase or decrease with respect of current dimension).

3. Compute the centroid inside new window.

4. If difference between old and new centroid coordinates is inferior to threshold, or if window is shifting outside picture edges, then stop. Else, compute new window dimensions and repeat from the first step.

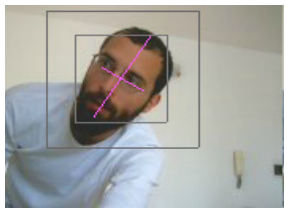

Fig. 7. Face tracking

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 (Pitas, 1993, Gonzalez, 2002).

Sobel was preferred as it is directional (the first matrix is row kernel S_x, the second one is column kernel S_y), thus permitting 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.

Once the image is transformed into binary space, integral projection is computed from each of them (Brunelli, 1993) as shown in Fig. 8. This returns vertical V(x) and horizontal H(y) occurrences histogram inside the ranges $x_1,\ldots,x_2$ and $y_1,\ldots,y_2$:

$$V(x)=\sum_{x=x_1}^{x_2} F'_x(x,y),\ H(y)=\sum_{y=y_1}^{y_2} F'_y(x,y), \tag{14}$$

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

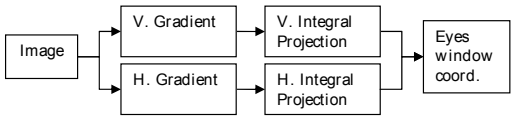

Fig. 8. Eyes window algorithm

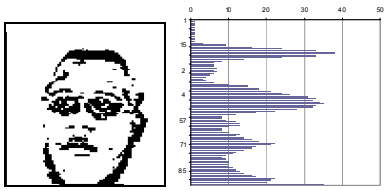

Fig. 9. Vertical gradient

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 finding the histogram peak around its 2/4 segment. Fig. 9 proves how features along vertical gradient are determined. If face axis inclination passes 9 degrees, integral projection does not return correct histogram. It is then necessary to rotate the image through coordinates changing matrix:

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

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 therefore possible to locate eye centres 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

$$E_M=\frac{1}{3}\left(C_b^2+\overline{C}_r^2+\frac{C_b}{C_r}\right), \tag{15}$$

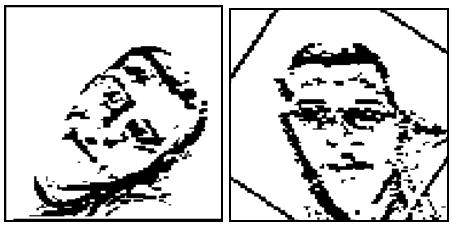

Fig. 10. Face inclination

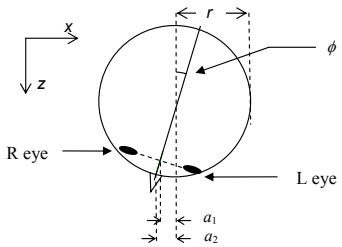

Fig. 11. Head pose

where colour components C_b^2, $\overline{C}_r^2$ and C_b/C_r are normalized to $\{0,\ldots,255\}$, and $\overline{C}_r=255-C_r$. This equation was built under the observation that regions colour components surrounding eyes have high blue values (C_b) and low red values (C_r) (Hsu, 2003). $E_M(x,y)$ grey scale 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:

$$\phi=\sin^{-1}(a/r), \tag{16}$$

where a is the distance between face centroid and face symmetry axis and r is head radius. Looking at Fig. 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 (Jie, 2000). 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 symmetry 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).

5. RESULTS

The system, implemented in C++, was tested in a computer with 256MB of memory and Intel Pentium 3 processor (1GHz). Results are obtained from processing over 2000 pictures as Fig. 12, with different faces in different positions. Still pictures were taken from a face database available in the Internet (Martinez, 1998).

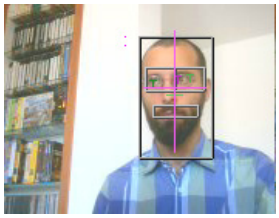

Fig. 12. Final result

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 1, 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 Fig. 13, each bar collecting results from a range of different face sizes.

Table 1. Performance summary

Res.	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

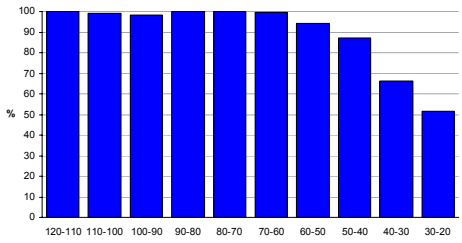

Fig. 13. Face detection success depending on face size; image: 176*x*144 pixels

Face detection success is almost independent on face roll θ. Measures were collected from images of constant distance between face and camera, with approximately 80×80 (±10) face size. Each element clusters a set of 20 degrees (±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. The graph in Fig. 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 head roll range was $\theta \in \{-30°, 30°\}$.

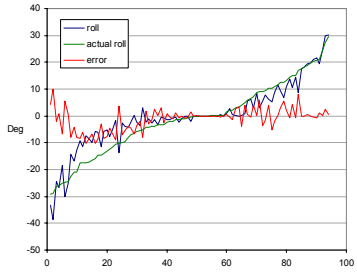

Fig. 14. Face roll measure

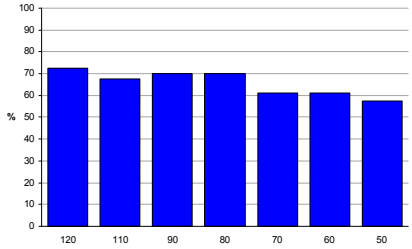

Fig. 15. Eyes detection success in different face sizes; image size: 176×144 pixels

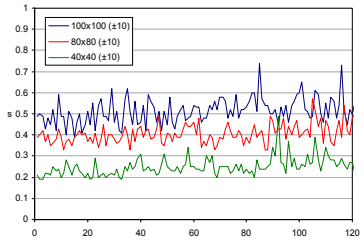

Fig. 16. Process time

Eyes detection is described in histogram of Fig. 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 Fig. 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 2. Detection rates: results comparisons (*depending on detection method)

Face size	This project	(Wu, 1999)	(Oliver, 2000)	(Hsu, 2003)
bigger than 100×120	98%	100%	96%	91%-99%*
bigger than 50×60	99%	97%		
bigger than 20×24	73%	100%		

Table 2 shows how this projects 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 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 and different measure and different hypothesis: for instance, looking at performance given by (Hsu, 2003) and (Oliver, 2000), face size range was not reported. Hence, it is not possible to compare this method's 73% rate with the percentage reported by (Oliver, 2000), as face size hypotheses are not the same.

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. Performance results are similar than in previous techniques, but with this approach computational cost and process time is reduced: skin region detection can reduce the search area by (in average) 42%. Therefore, fuzzy template matching could be applied to only 42% instead of 100% of the image area. This can be a starting point for face recognition or face controlled appliances.

REFERENCES

Lanitis A., C.J. Taylor, T.F. Cootes (1995). *An Automatic Face Identification System Using Flexible Appearance Models*. Image and Vision Computing, **vol. 13, no. 5**, pp. 393-401.

Yuille A., P. Hallinan, D. Cohen (1992). *Feature Extraction from Faces Using Deformable Templates*. Intl J. Computer Vision, **vol. 8, no. 2**, pp. 99-111.

Martinez A.M. and R. Benavente (1998). *The AR Face Database*. CVC Technical Report **no.24**.

Jie Huang F., T. Chen (2000). *Tracking of Multiple Faces for Human-Computer Interfaces and Virtual Environments*. Electrical and Computer Engineering.

Bradski G. R. (1998). *Computer Vision Face Tracking For Use in a Perceptual User Interface*. Intel Technology Journal **Q2 98**.

Wu H., Q. Chen, M. Yachida (1999). *Face Detection From Color Images Using a Fuzzy Pattern Matching Method*. IEEE transactions on pattern analysis and machine intelligence, **vol. 21, no. 6**.

Pitas I. (1993). *Digital Image processing Algorithms.* Prentice Hall.

Fan J., David. K. Y. Yau, Ahmed. K. Elmagarmid, Walid G. Aref (2001). *Automatic Image Segmentation by Integrating Color-Edge Extraction and Seeded Region Growing*. IEEE transactions on image processing, **vol. 10, no. 10**.

Chun J., K. Min, K. Park and J. 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.

Seo K., W. Kim, C. Oli, J. Lee (2002). *Face detection and facial feature extraction using colour Snake*. IEEE transactions on pattern analysis and machine intelligence.

Brunelli R., T. Poggio (1993). *Face recognition: features versus templates*. IEEE transactions on pattern analysis and machine intelligence.

Gonzalez R. C., R. E. Woods (2002). *Digital image processing*. Addison Wesley.

Hsu R. L., M. Abdel-Mottaleb, A. K. Jain (2003). *Face detection in color images*. IEEE transactions on pattern analysis and machine intelligence, **vol. 25, no. 12**.

Thilak Kumar R., S. Kumar Raja, A. G. Ramakrishnan (2001). *Eye detection using using color cues and projection functions*. IEEE transactions on pattern analysis and machine intelligence.

Phung S. L., A. Bouzerdoum, D. Chai (2001). *A novel skin color model in YCbCr color space and its application to human face detection*. IEEE transactions on pattern analysis and machine intelligence.

Pratt W. K. (1991). *Digital Image processing*. John Wiley & sons.

Appendix A. PROCESSING EXAMPLES

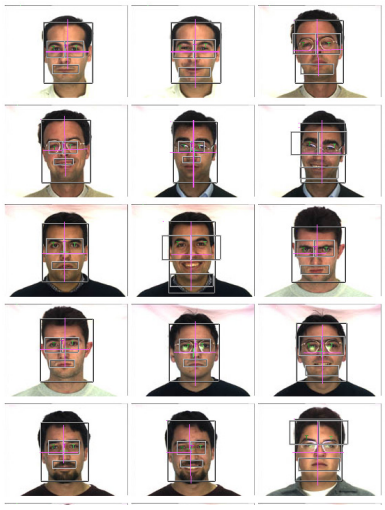

Fig. 17. Face detection with different faces.

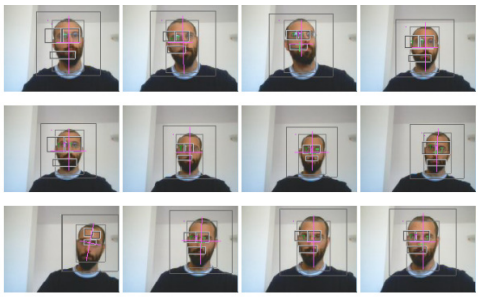

Fig. 18. Face tracking with video sequence.