Detection of frontal faces in video streams *

M. Castrillón Santana, J. Lorenzo Navarro, J. Cabrera Gámez, F.M. Hernández Tejera, J. Méndez Rodríguez **

Instituto Universitario de Sistemas Inteligentes y Aplicaciones Numéricas en Ingeniería (IUSIANI) - Universidad de Las Palmas de Gran Canaria - Edificio Central del Parque Científico Tecnológico - Campus Universitario de Tafira 35017 LAS PALMAS - SPAIN

Abstract This paper describes an approach for detection of frontal faces in real time (20-35Hz) for further processing. This approach makes use of a combination of previous detection tracking and color for selecting interest areas. On those areas, later facial features such as eyes, nose and mouth are searched based on geometric tests, appearance verification, temporal and spatial coherence. The system makes use of very simple techniques applied in a cascade approach, combined and coordinated with temporal information for improving performance. This module is a component of a complete system designed for detection, tracking and identification of individuals [1].

Keywords: Face detection, tracking, active vision, feature detection, HCI.

1 Introduction

Since its beginning, the evolution of human computer interaction (HCI) tools has been notorious and not trivial. Unfortunately, even today, accessing to these interaction tools requires training, as they are currently based on the use of devices that are clearly not natural for humans. Nowadays common interaction devices: mouse, keyboards and monitors are just, current technology artifacts. Oral communication plays a main role in human interaction, however we should not forget visual information such as body communication, gestures and facial expressions. On that context, it is easily observed that humans make simultaneous use of their motion, gesture abilities and sensing possibilities for communicating with their environment. If HCI gets closer to human communication schema, computer access would be wider and easier, making human computer interaction non-intrusive and more natural. For that reason a new trend of non intrusive interfaces, based on natural communication, is being developed using perceptual capabilities similar to humans [2].

Thus, we expect a computer to be able to detect a person in a non intrusive way. In this framework, Computer Vision capabilities can play a main role in

* Work partially funded by *Spanish Government and EU Project 1FD97-1580-C02-02 and Canary Islands Autonomous Government PI2000/048 research projects*

** Email:modesto@dis.ulpgc.es

HCI applications [2], such as handicapped assistants, augmented reality [3], and recent development of entertainment robots [4]. Among others these applications make use of this interaction abilities, presenting a challenging problem.

This paper describes a module for detecting frontal faces based on color, facial features detection and tracking of those features. The system has been developed with a major design requirement which is processing at frame rate video streams using standard hardware. Employing weak techniques, a set of heuristics and making use of temporal coherence, our current prototype presents promising results, allowing us to consider that this shuttle is affordable.

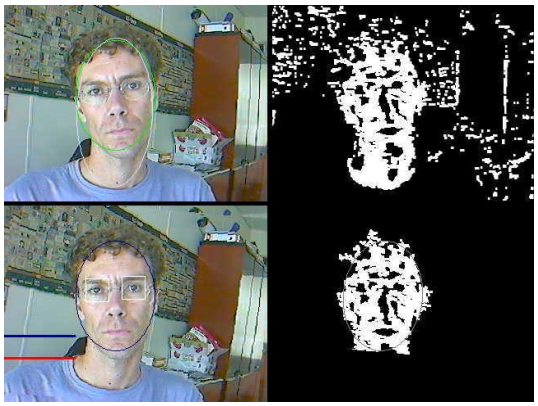

Figure 1. Ellipses fitted with and without neck pixels

2 Face detection

Face has been object of analysis by humans for centuries. Faces tell us who is a person or help us to guess features that are interesting for social interaction such as gender, age, expression and more. Face detection must be a necessary preprocessing step in any automatic face recognition [5] or face expression analyzer system [6]. However, the face detection problem commonly has not been considered in depth, perhaps due to it is just a previous step in a more categorical system (face recognition and facial expression). Even when many face detection methods have been proposed, it is only recently when Computer Vision community researchers have paid more attention to face detection problem, as recent surveys publication confirms [5,7].

Frontal faces are interesting due to the fact that most facial processing techniques such as recognition, facial expressions and more, make use of them. Those systems work under a standard size to reduce the problem dimensionality. Thus, any face detector system would finally transform properly the image to that work format, useful for face analysis tools.

Video streams consist of a rectangular image that could contain a set of potential areas of the image, which any of them could correspond to a human face. That is the main problem, for a face detector system, confirming/rejecting an area as a frontal face. In our framework we do not pretend to have a robustness such as the human system. Detecting any possible facial pose at any size seems to be an extremely hard problem and certainly not trivial, e.g. , a surveillance system can not expect that people show their faces clearly. Such a system must work continuously and should keep on looking at the person until he or she offers a good opportunity for the system to get a frontal view.

Different approaches described in [8,5,7] have been proposed to solve this problem: pattern recognition techniques, templates, neural networks and more. These systems are commonly compared using different datasets [5,7] which are composed of single images, not sequences. Most of these techniques were conceived for single images, performing an exhaustive search for restricted poses and sizes on the image. These approaches need a great computational effort, affecting seriously the performance of the system. As pointed out on those works, some information or invariant features, are available for improving performance. The authors refer to information such as color and motion as tools for optimizing any face detection algorithm. For example, color feature helps restricting search area, providing also the advantage of its orientation invariance, its robustness against scale changes, partial occlusion and its fast calculation making suitable for real time systems.

Our system will pay attention to color as it allows achieving real time restriction. However, it is well known that color is not robust under any circumstance.Many studies have located the skin color variance in a concrete area of a selected color space, however, when a light change appears this area seems to experiment a translation on color space. Thus, color perception can vary substantially for different environments (indoor, outdoor) specially when lighting conditions change [9]. This is color constancy problem.

2.1 Our color approach

Using a features selection approach [10] based on information theory, we have selected the color spaces that seems to get better discrimination performance. For our color experiment, 31752 color samples were used corresponding 17483 to non skin samples and the rest to skin samples. Color spaces studied were YUV, Normalized red and green $(\tilde{r}, \tilde{g})$ [11] , a Perceptual Uniform Color System $(Y_f U_f V_f)$ [12], RGB, $I_1 I_2 I_3$ being $(I_1 = \frac{r+g+b}{3}$, $I_2 = r - b$, $I_3 = g - \frac{r+b}{2})$.

GD measure [10] provided features sorted in relation to its discriminant features as follows: $\tilde{g}$, I_3, V_f, V, I_2, $\tilde{r}$, U_f, U, R, I_1, Y, Y_f, G and B. According to these results, the first color space completed in that classification is normalized red and green, $(\tilde{r}, \tilde{g})$. Thus, it was selected as the most discriminant color space and has been used in our work to define the skin color model by defining rectangular areas. This approach provides acceptable results within the context of our system without varying light conditions.

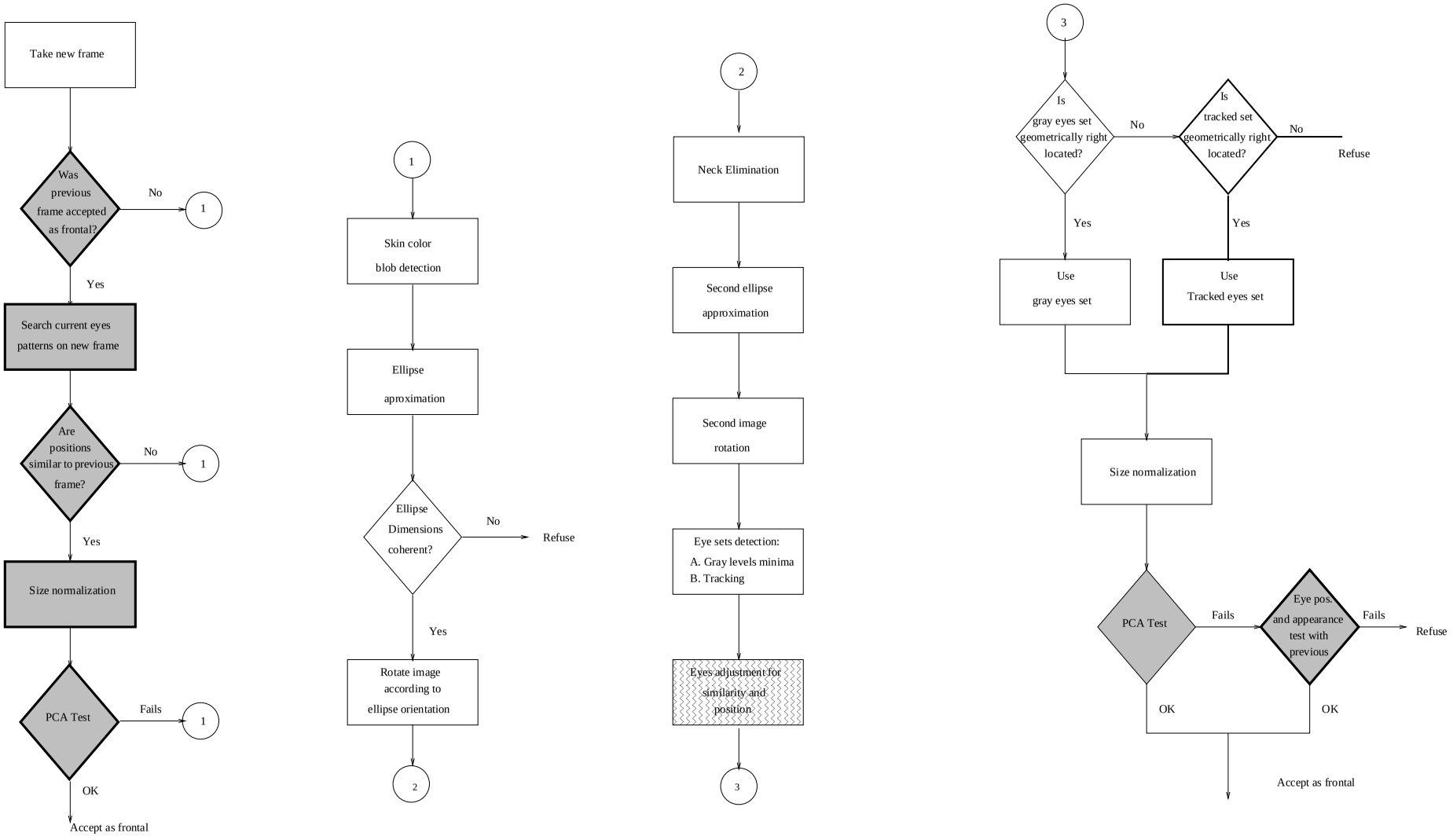

Figure 2. Detection process scheme

2.2 Features detection

Passive feature detection techniques have been treated extensively, adopting different schemas: gray level projections [13] and minima [14], static templates [15], snakes [16], eigenvectors [17], symmetry operators [18], morphological operators [19], Gabor filters [20], SVMs [21], etc, or even by hand allowing a better and more precise information specification.

Once a skin color blob has been detected, it is needed to determine the transformations to perform a normalization process to the standard size. This transformation would allow us to avoid differences that are not due to the individuals but the image taken, as for example the scale. The system locates certain key points or features on the faces. In our approach, pupils are detected using a combination of gray levels minima search and tracking. Their position on the image will define the transformation to apply. Also, if they are not detected, the system will consider that the image does not contain any frontal face.

3 The Procedure

The process, summarized in fig. 2, tries to detect first the potential eyes, and once they have been located proceeds performing some tests based on contextual knowledge about face geometry, appearance and temporal coherence in order to validate or refuse the hypothesis that eye positions recovered are coherent for a frontal view. Later that image is normalizes and a pattern recognition technique is applied to confirm it is a frontal face. In the following, the procedure is described briefly with some details:

Test with previous using tracking: This system is designed for processing video streams, thus before processing a new frame, previous frame result can be used. If previous frame was considered as frontal, in order to speedup the process (table 3), we perform a temporal coherence test. This test searches, in current frame, each eye on a window centered in previous detected eye position with a dimension of $0.6 * inter - eyes - distance$ (using the schema described below and deeply in [22]). If the position detected is similar to the previous one (for both eyes), the system performs a rotation, later a normalization and finally applies the PCA test (see below). If the image passes these tests, it is considered as frontal. In any other case, normal eyes detection procedure is carried out.

Color Blob Detection and Ellipse Approximation: As we mentioned, normalized red and green color space [11] is used for skin color detection. Those blobs classified as skin colored are fitted to a general ellipse using the technique described in [14]. Some ellipses are rejected using geometric filters: 1) Those considered too big (according to the image), 2) those too small with short ellipse axis under 15 pixels, 3) Those whose vertical axis is not larger (as we expect faces almost in upright position), and 4) those with unexpected axis ratio.

Figure 3. Detection example. Top left: Input image with first ellipse. Bottom left: Second ellipse and search areas for eyes. Top right: Face with features painted. Bottom right: Last face detected, and first face detected and experiment average face.

Face Orientation: Ellipse calculation also provides an orientation for the face. The orientation obtained is employed for rotating the source image in order to get a face image where both eyes lie on a horizontal line.

Neck Elimination: Clothes and hair styles affect the shape of the blob. A face color blob could contain the neck enlarging the blob, thus, face geometric knowledge and heuristics are used to eliminate those blob pixels that are not part of the face, as for example neck, fig. 1. Finally a new ellipse is approximated.

Eyes Detection: Once the neck is avoided, the ellipse is well fitted to the face. Faces present geometric relations for features positions, thus, we can search for eyes in a coherent manner, where eyes should be for a frontal face. In a general case, some standard dimensions are used for this topic, but if we pay attention to information provided by video streams, we could make use of good eyes position detection obtained from last correct detection, referred to skin blob center. Once a user is being detected, search areas are adapted and restricted to his/her dimensions. Losing the subject (no detection in a number of consecutive frames) will force the system to reset search areas. In this step, two eye sets of candidates are obtained, one using gray levels, and another tracking eyes based on difference images [22]. Tracking eyes mechanism makes use of a couple of thresholds: 1) a fix threshold, $threshold_{lost}$, for determining a no detection, i. e., a pattern lost, and 2) an adaptable threshold, $threshold_{update}$, that will determine eye pattern update ($threshold_{update} < threshold_{low}$). A brief description of tracking mechanism is as follows:

Compute Difference Image: A pattern ($16x16$) is searched on a window (wxw) of interest. This window is centered on last valid detection, with dimensions related with short ellipse axis (0.6*shortaxis). On each pixel, it is computed:

$$\sum_{i=1}^{w}\sum_{j=1}^{w} abs(Pattern(i,j) - Image(lasty + i - w/2, lastx + j - w/2)) \quad (1)$$

Minima Search: On that window, the minimum value is selected as the most similar to previous eye pattern.

Check Minimum Value: If minimum value is greater than a threshold, $threshold_{lost}$, the pattern is considered lost.

Second Minima Search: Second minima on that window is searched.

Update: If the minimum value is lower than $threshold_{lost}$, but greater than $threshold_{update}$, or if $threshold_{update}$ is greater than the second minima, the pattern is updated and $threshold_{update}$ forced to be smaller than second minima.

Too Close Eyes Adjustment: On each candidate sets, if eyes are detected too close in relation to ellipse dimensions, the closest to ellipse center is rejected and searched again avoiding the area where it was detected. This test helps when wearing glasses as the glasses elements could be darker.

Eyes adjustment by image difference: Given both possible eyes sets, a small subimage is centered in the one with the lowest gray level. This subimage is compared with subimages centered in a window around the other possible eye, selecting as candidate the position with less difference (computed analogous to eq. 1) as eyes should be similar in appearance.

Figure 4. Input image and right eye (in the image) zoom. The darkest point is not the iris center.

Geometric tests: Some tests are applied first to gray level eyes set, in case of failure, tracked eyes (obtained by previous frames info) set is then submitted:

1.- Intereyes distance test: Eyes should be at a certain distance coherent with ellipse dimension. This distance should be greater that a measure defined by ellipse dimensions, $ellipse_short_axis * 0.75$ and lower than another ratio according to ellipse dimensions $shortaxis * 1.5$.

2.- **Horizontal test:** As we mentioned, resulting eye candidates should lie almost on a horizontal line if ellipse orientation is correct. Using a threshold adapted to ellipse dimension, $shortaxis/6.0+0.5$, we are refusing candidate eyes that are too far from an horizontal line. For eyes that are almost horizontal but not completely, the image is rotated one more time, to force eyes to be on the same row.

3.- **Lateral eye position test:** Eyes positions could provide a clue of a lateral view. Face position is considered lateral if the distance from eyes to the closest border of the ellipse differs considerably.

Normalization: A candidate set that verifies all the previous requirements is then scaled and translated to fit a standard position and size. Over this standard size face image we make use of an ellipse for removing hair and background areas.

PCA test: A final test applied in order to reduce false positives makes use of reconstruction error [23] obtained after projecting the normalized face in PCA eigenspace. This schema provides better results than comparing with an average face or a session average face, reducing the number of false positives (see table 3). For calculating the PCA decomposition a small set of 16 images (none taken from sample sequences, nor the same subject and camera).

The face is considered frontal: In that case some actions are taken:

Mouth and Nose Detection: Once we have detected eyes and they are in horizontal relative position, we search down according to intereyes distance for a horizontal area with low gray levels.

Eyes patterns update: As the face has been considered frontal, we update eye patterns (if necessary according to tracking mechanism) to use them for detecting eyes with correlation.

Tests included	Frontal seq. A	Frontal seq. B	Time (PIII 1GHz)
Basic implementation	281/251	310/291	48 ms.
Prev. plus PCA Test	275/251	291/289	50 ms
Prev. plus adaptive eye search	373/358	335/319	54 ms
Prev. plus tracked eye set	424/408	375/343	54 ms.
Prev. plus testing previous frame for frontal detection	423/403	395/365	27 ms.

Table 1. Results achieved and processing time integrating different modules (fig. 2)

4 Detection Experiments

The experiments have been carried out over two sequences (Ground data for pupil position and sequences are available at [24]). Both sequences were taken

in different days, in an indoor scenario, using general purpose hardware (web-cams), without any controlled nor artificial illumination. The user was sat in front of the computer posing (i.e., also with out of plane rotations) in sequence A and speaking in sequence B. Both sequences contain 450 frames of 320x240 each, corresponding to 30 seconds (15Hz). This implementation, developed using OpenCV library.

Approach	Frontal faces detected Seq. A	Frontal faces detected Seq. B
Rowley NNBFD	361	449
ENCARA	423/403	395/365

Table 2. Frontal face detections comparison using Dr. Rowley's method and ENCARA

In table 3, we expose some results achieved with different implementations of the system. These results seem to justify the benefits of the integration of several techniques in a cascade and cooperative approach. In order to analyze the goodness for each implementation, on each field two numbers are reported: 1) the number of frontal faces returned, and 2) those whose eyes were considered close enough to eyes marked by hand (Automatic position was compared with $euclidean - eye - distance/8$). The basic implementation just searches eyes in a restricted area of color blob attending to gray levels applying geometric test to decide if a candidate is a frontal face. Second implementation, integrates a reconstruction error test based on PCA, reducing lightly the false positives ratio (fill pattern on last column in fig. 2). Third implementation performs a more intelligent eye candidates, adjusting eyes when they are too close and/or using image differences (zigzag fill pattern in fig. 2). Fourth implementation introduces also the computation of a eye set using the tracking mechanism with last known pattern rectangle (thick lines in fig. 2), improving detection rate. With this implementation it is achieved a rate of almost 18.5 Hz, which could be adopted for weak real time tasks. Finally, last implementation focuses increasing rate without decreasing performance. In order to speed up the process, an explicit use of coherence is performed if immediate previous frame was selected as frontal. In that case, those eyes are tracked and new eye position compared with previous (thick lines and fill pattern fig. 2)). This implementation increases the rate, up to 37Hz without affecting system performance. Certainly, in this implementation, rate increase depends on the sequence. In our examples, processing at frame rate, as the camera provides frames at 15Hz, thus, in 1 second the system would need just 405 ms. for processing those frames, giving time slices for other tasks. This final implementation would be referred as ENCARA.

For both sequences, it was also employed a robust face detector due to Dr. Rowley [8], to provide a reference to verify results achieved with ENCARA. Dr. Rowley's Neural Network based Face Detection (NNBFD) technique, does not depend on blob size as it processes the whole image, performing slower (700-800

ms. per frame). For Rowley's method, we report only the number of frontal faces detected, as the method does not provide always eye positions. According to the comparison betweeen both systems, table 4, ENCARA presents a promising behavior in this context.

Sequence	Frontal detection	Right eyes detection	Both eyes too distant	Right eye too distant	Left eye too distant	Right av. error	Left av. error
Seq. A	423	403	14	5	1	(2,1)	(2,1)
Seq. B	395	365	2	10	18	(2,1)	(2,1)

Table 3. Eye position detection error summary using ENCARA.

Finally, table 4 presents a summary of eye detection error according to ground data, i. e., false positives detections, using ENCARA. Before further comment we should mention that in both sequences after those false eye detections, ENCARA was able to detect them properly again. For the first sequence, we can observe that 14 times both eye positions are refused. Observing the scene, it happens when the subject, who wears glasses, presents a lateral pose and glasses junctions are confused with pupils. Testing those image with PCA test, was not enough to refuse them as they were face lateral views, but with incorrect eye detections. We plan to integrate, in short term, lateral views detection mechanisms in order to avoid taking incorrect eye candidates. For the second sequence, the wrong detection peak is observed with left eye. Analyzing the video, it is observed that eyebrow was confused with eye for those frames. The integration of an eye appearance filter, and not only a whole face appearance one, should be studied. Positional average error computed for eye positions is presented in last two columns in table 4. These results provide a similar error for both sequences. In fig. 4, it is presented an typical sequence frame, average iris diameter is 8 pixels. It should be noted that ENCARA is initialized (and recovered) using a gray level technique and obviously it is not assured that an image will reflect the iris center as the darkest point. That reason could explain a greater error in x.

5 Conclusions and Future work

The module described in this paper, is conceived to be used with general purpose hardware for real-time processing. It is designed in a cascade and cooperative fashion of opportunistic classifiers to confirm/reject the frontal view hypothesis. It uses opportunities presented to the system, making use of a set of simple and known tools, but weak by themselves. According to experiments system performance increases when combining simple techniques. It is important to take note of the results achieved for the combination tools trying to exploit in a first step information that is enclosed in a video stream. In the current implementation, position and appearance of the facial features considered (eyes)

are used in following frames once we got an initial good detection, achieving in our experiments a 37Hz rate.

Our two main goals are real time and robustness. ENCARA needs a long path to walk before reaching robustness goal, but we expect to get more from temporal coherence. Our immediate work will be testing more sequences with different subjects and illumination conditions. More exhaustive experiments must be done in order to confirm the promising performance of ENCARA for caucasian individuals. However, we miss in the community test sequences for comparison of different techniques. For our sequences, the same color area was searched on both, but it is evident that this color area will not be robust enough for illumination changes. A proper color model, in our experiments, produces a performance comparable to Rowley's method, but at greater rate. Skin color model affects initial localization, thus affecting system effectiveness. We expect to focus on color update, making use of information available, examining surround areas of detected eyes, updating color model within skin locus [25]. Also, next steps will afford to provide more robustness to the module adding tools to be aware of lateral views in the process. Also, we expect to make use of mouth and nose detection in order to combine evidences and allowing the system to be more robust against a feature lost.

6 Acknowledgements

We would like to thank to Dr. Henry Rowley, Robotics Institute of Carnegie Mellon University, for providing the code of their face detector for comparison purposes.

References

1. O. Déniz, M. Castrillón, J. Lorenzo, and M. Hernández. An incremental learning algorithm for face recognition. In *Post-ECCV Workshop on Biometric Authentication*, 2002. Copenhagen, Denmark.
2. Alex Pentland. Looking at people: Sensing for ubiquitous and wearable computing. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, January 2000.
3. Bernt Schiele, Tony Jebara, and Nuria Oliver. Sensory augmented computing: Wearing the museum's guide. *IEEE Micro*, May/June 2001.
4. Sebastian Thrun, Mareb Bennewitz, Wolfram Burgard, Armin B. Cremers, Frank Dellaert, Dieter Fox, Dirk Hähnel, Charles Rosenberg, Nicholas Roy, Jamieson Schulte, and Dirk Schulz. Minerva: A second-generation museum tour-guide robot. Technical report, Carnegie Mellon University, 1998.
5. E. Hjelmas and B. K. Low. Face detection: A survey. *Computer Vision and Image Understanding*, 83(3), 2001. Erik Hjelmas and Boon Kee Low.
6. Paul Ekman and Erika Rosenberg. *What the Face Reveals : Basic and Applied Studies of Spontaneous Expression Using the Facial Action Coding System (Facs).* Series in Affective Science. Oxford University Press, 1998.
7. Ming-Hsuan Yang, David Kriegman, and Narendra Ahuja. Detecting faces in images: A survey. *Transactions on Pattern Analysis and Machine Intelligence*, 24(1):34–58, 2002.

8. Henry A. Rowley. *Neural Network-Based Face Detection.* PhD thesis, Carnegie Mellon University, May 1999.
9. Moritz Störring, Hans J. Andersen, and Erik Granum. Skin colour detection under changing lighting conditions. In *7th Symposium on Intelligent Robotics Systems*, July 1999.
10. J. Lorenzo, M. Hernández, and J. Méndez. Gd: A measure based on information theory for attribute selection. *Lectures Notes in Artificial Intelligence*, 1484:124–135, 1998.
11. Christopher Wren, Ali Azarrbayejani, Trevor Darrell, and Alex Pentland. Pfinder: Real-time tracking of the human body. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 19(7), July 1997.
12. Haiyuan Wu, Qian Chen, and Masahiko Yachida. Face detection from color images using a fuzzy pattern matching method. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 21(6), June 1999.
13. R. Brunelli and T. Poggio. Face recognition: Features versus templates. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 15(10):1042–1052, 1993.
14. Karin Sobottka and Ioannis Pitas. A novel method for automatic face segmentation, face feature extraction and tracking. *Signal Processing: Image Communication*, 12(3), 1998.
15. C. Wong, D. Kortenkamp, and M. Speich. A mobile robot that recognizes people. *IEEE International Conference on Tools and Artificial intelligence*, 1995.
16. Andreas Lanitis, Chris Taylor, and Timothy F.Cootes. Automatic interpretation and coding of face image using flexible models. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 19(7), July 1997.
17. Alex Pentland, Baback Moghaddam, and Thad Starner. View based and modular eigenspaces for face recognition. *Proc. IEEE Conference on CVPR'94*, 1994.
18. Daniel Reisfeld and Yehezkel Yeshurun. Preprocessing of face images: Detection of features and pose normalization. *Computer Vision and Image Understanding*, 71(3), September 1998.
19. M. Tistarelli and E. Grosso. Active vision-based face authentication. *Image and Vision Computing*, 18, 2000.
20. F. Smeraldi, O. Carmona, and J. Bigün. Saccadic search with gabor features applied to eye detection and real-time head tracking. *Image and Vision Computing*, 18, 2000.
21. Jeffrey Huang, David Ii, Xuhui Shao, and Harry Wechsler. Pose discrimination and eye detection using support vector machines. In *Proc. NATO-ASI on Face Recognition: From Theory to Applications*, 1998.
22. M. Hernández, J. Cabrera, M. Castrillón, A .C Domínguez, C. Guerra, D. Hernández, and J. Isern. Deseo: An active vision system for detection, tracking and recognition. *Lecture Notes on Computer Science, 1542*, 1999. Springer-Verlag, ICVS'99, Gran Canaria.
23. Erik Hjelmas and Ivar Farup. Experimental comparison of face/non-face classifiers. In *Procs. of the Third International Conference on Audio- and Video-Based Person Authentication. Lecture Notes in Computer Science 2091*, 2001.
24. M. Castrillón Santana. Face sequences dataset, http://kassandra.techfak/uni-bielefeld.de/euron, 2000.
25. M. Soriano, B. Martinkauppi, S. Huovinen, and M. Laaksonen. Using the skin locus to cope with changing illumination conditions in color-based face tracking. In *Proc. Nordic Signal Processing Symposium (NORSIG2000)*, pages 383–386, June 13-15 2000. Kolmården, Sweden.