Real-Time Detection of Human Faces in Uncontrolled Environments

Ara V. Nefian
Georgia Institute of Technology, Department of Electrical Engineering
Atlanta, Georgia , 30332

Mehdi Khosravi
NCR Human Interface Technology Center
Atlanta, Georgia, 30309

Monson H. Hayes
Georgia Institute of Technology, Department of Electrical Engineering
Atlanta, Georgia, 30332

ABSTRACT

This paper presents an approach for the detection of human face and eyes in real time and in uncontrolled environments. The system has been implemented on a PC platform with the aid of simple commercial devices such as an NTSC video camera and a monochrome frame grabber. The approach is based on a probabilistic framework that uses a deformable template model to describe the human face. The system has been tested on both head-and-shoulder sequences as well as complex scenes with multiple people and random motion. The system is able to locate the eyes from different head poses (rotations in image plane as well as in depth). The information provided by the location of the eyes is used to extract faces with frontal pose from a video sequence. The extracted frontal frames can be passed to recognition and classification systems for further processing.

Keywords : Face Detection, Eye Detection, Face Segmentation, Ellipse Fitting

1. INTRODUCTION

In recent years, face detection from video data has become a popular research area. There are numerous commercial applications of face detection in face recognition, verification, classification, identification as well as security access and multimedia. To extract the human faces in an uncontrolled environment most of these applications must deal with the difficult problems of variations in lighting, variations in pose, occlusion of people by other people, and cluttered or non-uniform backgrounds.

A review of the approaches to face detection that have been proposed are described in[1]. In [2], Sung and Poggio presented an example-based learning approach for locating unoccluded human frontal faces. The approach measures a distance between the local image and a few view-based “face” and “non face” pattern prototypes at each image location to locate the face. In [3], Turk and Pentland used the distance to a “face space“, defined by “eigenfaces”, to locate and track frontal human faces. In [4], human faces were detected by searching for significant facial features at each location in the image. In [5] and [6] a deformable template based approach was used to detect faces and to extract facial features.

In this paper we present a probabilistic framework for face detection . The goal of our system is to efficiently segment human faces, independent of their size and orientation, from a known but uncontrolled background. A deformable template-based model has been used to describe the human face. An eye detection module processes the extracted faces to filter the frontal faces for further processing. The approach that we have used performs well, in crowded scenes where human bodies occlude each other.

2. SYSTEM OVERVIEW

In the first stage of our system, a statistical model for the background is created, and connected pixels with large deviations from this model are assigned to the foreground. The output of this stage is a set of connected foreground pixels that determine the foreground regions.

The foreground regions are analyzed in more detail to detect the head. It is known that if there is only one head in the image, then it may be detected by finding the upper region in each set of connected foreground regions [3]. However, this technique fails when people in an image are occluded by other people. In this case, a foreground region may correspond to two or more people, and finding the regions corresponding to heads requires a more complicated approach. This paper considers the case of partial people occlusion in which bodies are occluded by other bodies, but heads are not occluded. To determine the head positions, it is natural to determine the number of people in each foreground region. N separate models λ_i, $i = 1\ldots, N$ are built, each model λ_i corresponding to i people in a set of connected foreground region. Based on the assumption that faces are vertical and are not occluded, the model parameters for model λ_i are $(x_0, x_1, \ldots x_i)$ where i is the number of people and the $x_k, k = 1,..,i$ specify the horizontal coordinates of the vertical boundaries that separate the i head regions in the model λ_i. The approach used to determine the number of people in each foreground region is to select the model λ_i for which the maximum likelihood is achieved.

$$\lambda = \underset{i \in [1,N]}{\arg\max}\, P(O(x,y)|\lambda_i) \qquad (1)$$

where the observations $O(x,y)$ are the pixel intensities at coordinates (x,y) in the foreground regions and $P(O(x,y)|\lambda_i)$ is the likelihood function for the i^{th} model.

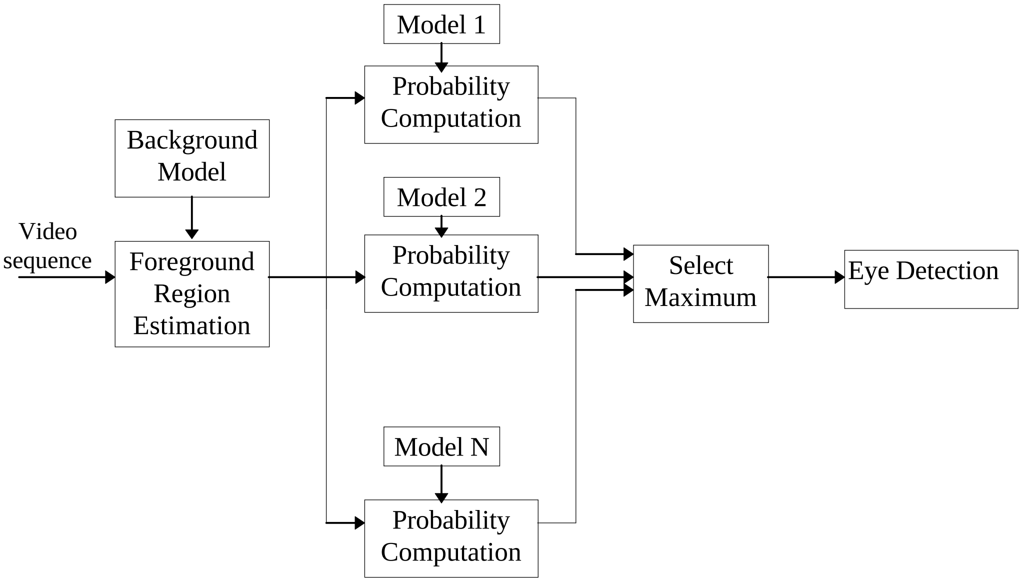


Figure 1. Face and Eye detection system

The probability computation step in Figure 1 determines the likelihood functions for each model. In this stage the observations $O(x,y)$ in the foreground regions are used to find for each model λ_i the optimal set of parameters

$(x_0, x_1, \ldots x_i)$ that maximize $P(O(x,y)|\lambda_i)$, i.e. to find the parameters $(x_0, x_1, \ldots x_i)$ that "best" segment the foreground regions. It will be shown later in this paper that the computation of $P(O(x,y)|\lambda_i)$ for each set of model parameters requires an efficient head detection algorithm inside each rectangular window bordered by x_{j-1} and x_j, $j = 1,..,i$. It is common to approximate the support of the human face by an ellipse. In addition, we have found that the ellipse aspect ratio of the human face is, for many situations, invariant to rotations in the image plane as well as rotations in depth. Based on the above, the head model is parametrized by the set (x_0, y_0, a, b) where x_0 and y_0, are the coordinates of the ellipse centroid and a and b are the axis of the ellipse. The set (x_0, y_0, a, b) is determined through an efficient ellipse fitting algorithm that will be descried later in this paper. The head model can be more complex, and can include information from the eyes as well as other facial features. Considering a simpler model for the human face may lead to the detection of objects that do not correspond to a face. These objects are removed in the next stage of our system by running an efficient eye detection algorithm.

3. SEGMENTATION OF FOREGROUND REGIONS

To extract the moving objects, the background is modeled as a texture with the intensity of each point modeled by a Gaussian distribution with mean μ and variance σ, $N_b(\mu,\sigma)$. The pixels in the image are classified as foreground if $p(O(x,y)|N_b(\mu,\sigma)) \leq T$ and as background if $p(O(x,y)|N_b(\mu,\sigma)) > T$. The observation $O(x,y)$ represents the intensity of the pixels at location (x,y). T is a constant threshold.

The connectivity analysis of the "foreground" pixels generates connected sets of pixels, i.e. sets of pixels that are adjacent or touching. Each of the above sets of pixels describe a foreground region. Small foreground regions are assumed to be due to shadow, camera noise and lighting variations and are removed. The approach used to foreground regions segmentation illustrated in Figure 2.

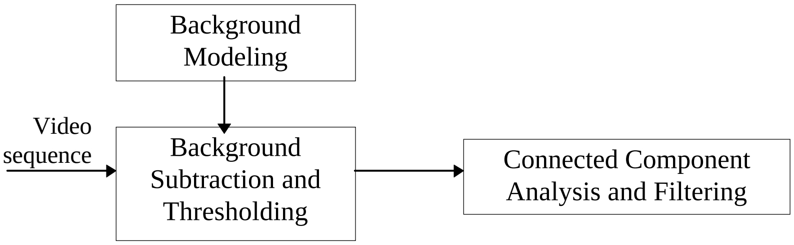

Background Modeling
Video sequence
Background Subtraction and Thresholding
Connected Component Analysis and Filtering

Figure 2. Foreground regions segmentation

4. COMPUTATION OF MODEL LIKELIHOOD FUNCTIONS

Based on the assumption that faces are vertical and are not occluded, we found it appropriate to parametrize models λ_i over the set of parameters $(x_0, x_1, \ldots x_i)$ which are the horizontal coordinates of the vertical borders that separate individual faces in each foreground region. The set of parameters $(x_0, x_1, \ldots x_i)$ is computed iteratively to maximize $P(O(x,y)|\lambda_i)$. In a HMM implementation this corresponds to the training phase in which the model parameters are optimized to best describe the observed data [9].

To define the likelihood functions $P(O(x,y)|\lambda_i)$, a preliminary discussion about the head detection algorithm is necessary. In this paper, the head is determined by fitting an ellipse around the upper portions of the foreground regions inside each area bounded by x_{j-1}, x_j $j = 1,..,i$. In fact the head detection problem is reduced to finding the set of parameters

(x_0, y_0, a, b) that describe an ellipse type deformable template. Parameters x_0 and y_0 describe the ellipse centroid coordinates and a and b are the ellipse axis. The head detection algorithm will be described in more detail in the next section. For each set of parameters (x_0, y_0, a, b) a rectangular template is defined by the set of parameters $(x_0, y_0, \alpha a, \alpha b)$, where x_0 and y_0 are the coordinates of the center of the rectangle and $\alpha a, \alpha b$ are the width and length of the rectangle, and α is some constant. In each area bounded by x_{j-1}, x_j, R_{0j} is the set of pixels outside the ellipse template and inside the rectangle template and R_{1j} is the set of pixels inside the ellipse template. The regions R_{0j} and R_{1j} locally classify the image in "face" and "not face" regions. Based on the above discussion the likelihood function $P(O(x,y)|\lambda_i)$ for the model λ_i is determined by the ratio of the number of foreground pixels classified as "face" and background pixels classified as "non face" in each area bounded by x_{j-1}, x_j, $j = 1,..,i$, over the total number of pixels in "face" and "not face" regions.

$$P(O(x,y)|\lambda_i) = \frac{\sum_{j=1}^{i}\left(\sum_{(x,y)\in R_{0j}} f(O(x,y)) + \sum_{(x,y)\in R_{1j}} b(O(x,y))\right)}{\sum_{j=1}^{i}\left(\sum_{(x,y)\in R_{0j}} b(O(x,y)) + \sum_{(x,y)\in R_{0j}} f(O(x,y)) + \sum_{(x,y)\in R_{1j}} b(O(x,y)) + \sum_{(x,y)\in R_{1j}} f(O(x,y))\right)} \tag{3}$$

where

$$b(O(x,y)) = \begin{cases} 1, & \text{if } p(O(x,y)|N_b(\mu,\sigma)) > T \\ 0, & \text{otherwise} \end{cases} \tag{4}$$

and

$$f(O(x,y)) = \begin{cases} 1, & \text{if } p(O(x,y)|N_b(\mu,\sigma)) < T \\ 0, & \text{otherwise} \end{cases} \tag{5}$$

The goal in this section is not only to compute the likelihood functions $P(O(x,y)|\lambda_i)$ for a set of parameters $(x_0, x_1, \ldots x_i)$, but also to determine the set of parameters that maximize $P(O(x,y)|\lambda_i)$. The initial parameters $(x_0, x_1, \ldots x_i)$ for model λ_i are chosen to uniformly segment the data i.e. $x_j - x_{j-1} = (x_i - x_0)/i$, $j = 1,..,i$. The parameters $(x_0, x_1, \ldots x_i)$ are iteratively adjusted to maximize $P(O(x,y)|\lambda_i)$, as illustrated in Figure.3. The iterations are terminated if the difference of the likelihood functions in two consecutive iterations is smaller than a threshold.

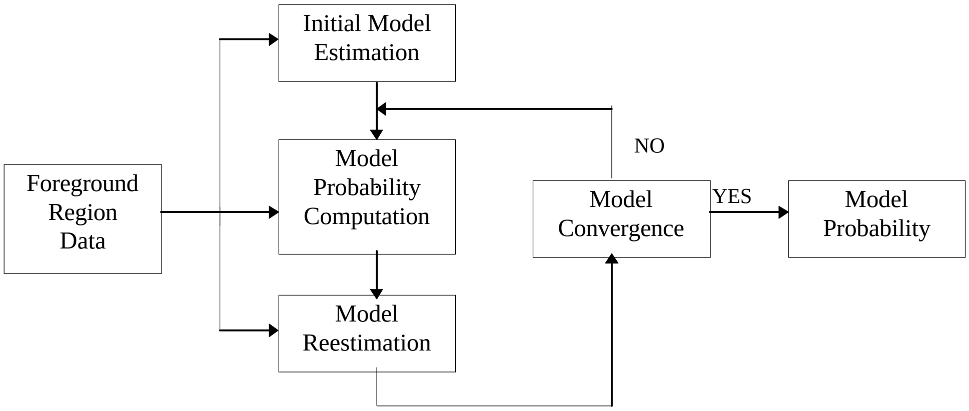


Figure 3. Computation of Model Likelihood

5. THE HEAD DETECTION ALGORITHM

The head is detected by fitting an ellipse around the upper portion of the foreground region inside the area bounded by x_{j-1}, x_j for $j = 1,..,i$. The objective in an ellipse fitting algorithm is to find the x_0, y_0, a and b parameters of the ellipse such that:

$$((x - x_0) / a)^2 + ((y - y_0) / b)^2 = 1 \tag{6}$$

A general technique for fitting the ellipse is the Hough Transform. However, the computational complexity of this approach as well as the need for a robust edge detection algorithm make it ineffective for real-time applications.

Our approach for fitting the ellipse is an inexpensive recursive technique that reduces the search for the ellipse parameters from a four dimensional space (x_0, y_0, a, b) to a one dimensional one. The parameter space of the ellipse is reduced based on the following observations:

1. The width of the ellipse at iteration $k + 1$ is equal to the distance between the right most and left most point of the foreground region at the line corresponding to the current centroid position, $y_0^{(k)}$ i.e.

$$a^{(k+1)} = f_1(y_0^{(k)}) . \tag{7}$$

where function f_1 is determined by the boundary of the foreground region.

2. The centroid of the ellipse is located on the so called vertical skeleton of the region representing the person. The vertical skeleton is computed by taking the middle point between the left most and the right most points for each line of the region. The $x_0^{(k+1)}$ coordinate of the centroid of the ellipse at iteration $k + 1$ is located on the vertical skeleton at the line $y_0^{(k)}$ corresponding to the current centroid position. Hence $x_0^{(k+1)}$ will be uniquely determined as a function of $y_0^{(k)}$.

$$x_0^{(k+1)} = f_2(y_0^{(k)}) , \tag{8}$$

where function f_2 is a function determined by the vertical skeleton.

3. The b parameter of the ellipse (the height) is generally very difficult to obtain with high accuracy due to the difficulties in finding the chin line. However, generally the height to width ratio of the ellipse can be considered to be a constant,
such as M . Then, from Equation (7)

$$b^{(k+1)} = Ma^{(k+1)} = Mf_2(y_0^{(k)}) \tag{9}$$

From (6) we write

$$y_0^{(k+1)} = F(x_0^{(k+1)}, a^{(k+1)}, b^{(k+1)}) \tag{10}$$

Equations (7), (8) ,(9) and (10) lead to:

$$y_0^{(k+1)} = G(y_0^{(k)}) , \tag{11}$$

which describes the iterative ellipse-fitting algorithm. Equation (11) indicates that we have reduced the four dimensional problem of finding the ellipse parameters to an implicit equation with one unknown y_0 .

We stop the iterations when the distance between two consecutive centroids is smaller than a threshold. When the iterations stop x_0, y_0, a and b describe the four parameters of the ellipse. The initial y-coordinate of the ellipse centroid, $y_0^{(0)}$ has to be chosen close enough to the top of the object on the vertical skeleton in order for the algorithm to perform well for all types of sequences from head-and-shoulder to full-body sequences.

5.1. Convergence of Head Detection Algorithm

In the previous section we have shown that y_0 can be determined iteratively from Equation (11). Also we have shown that the knowledge of y_0 is enough to determine the ellipse parameters. The head detection algorithm is illustrated in Figure 4.

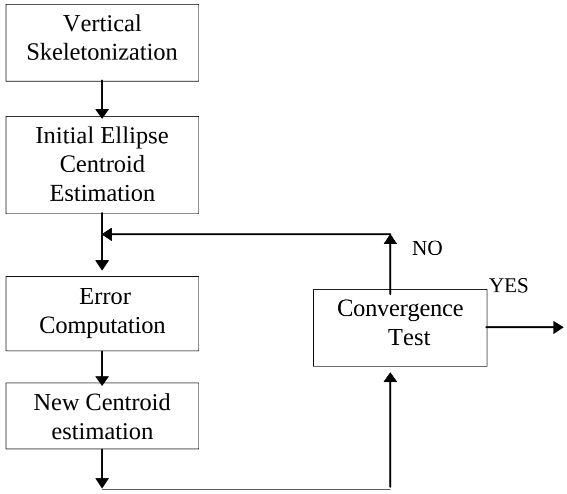


Figure 4. Ellipse Fitting Algorithm

In order to find the solution for equation (11) a linear estimate for $y_0^{(k+1)}$ is derived:

$$y_0^{(k+1)} = y_0^{(k)} + \mu e(k) \tag{12}$$

where,

$$e(k) = b^{(k)} - Ma^{(k)} \tag{13}$$

We will show in this section that the above iterations converge to the ellipse parameters for an ellipse type contour. From equation (6), the distance between the right most and left most point of the ellipse corresponding to $y_0^{(k)}$ is determined by:

$$a^{(k)} = 2a\sqrt{1-((y_0^{(k)} - y_0) / Ma)^2} \tag{14}$$

and the distance between the top of the ellipse and $y_0^{(k)}$ is determined by

$$b^{(k)} = y_0 + Ma - y_0^{(k)} \tag{15}$$

Hence, for $\mu = 1$, equation (6) becomes:

$$y_0^{(k+1)} - y_0 = Ma - Ma\sqrt{1-((y_0^{(k)} - y_0) / Ma)^2} \tag{16}$$

From the above equation it can be proved that

$$|y_0^{(k+1)} - y_0|^2 < |y_0^{(k)} - y_0|^2 \tag{17}$$

for any $y_0^{(k)}$ for which $|y_0^{(k)} - y_0| < Ma$. This shows that the recurrence defined in equation (12) converges to y_0.

6. EYE DETECTION

The detected ellipses are potentially the region of support for human faces. After the detection of these regions, a more refined model for the face is required in order to determine which of the detected regions in previous stages correspond to valid faces. The use of an eye detection algorithm in conjunction with the head detection module improves the accuracy of the head model and removes regions corresponding to back views of faces or other regions that do not correspond to a face. Eye detection results can also be used to estimate the face pose and to determine the image containing the most frontal pose among a sequence of images. This result may then be used in recognition and classification systems.

In previous work [1], eyes were successfully detected from frontal views. For frontal views, eye detection that is based on geometrical measures was extensively studied [7,8]. In [6], Yuilee, Hallinan and Cohendeveloped a deformable template based approach to facial feature detection. However, these methods may lead to problems in the analysis of profile or back views. Moreover, the assumption of dealing with frontal faces is not valid for real world applications.

This paper uses an eye detection algorithm based on both region size and geometrical measure filtering. The exclusive use of geometrical measures to detect the eyes inside a rectangular window around the ellipse centroid (eye band) may lead to problems in the analysis of non-frontal faces. In these cases, the hair regions inside the eye band generate small hair regions that are not connected each to other and that are in general close in size and intensity to the eye regions. Under the assumption of varying poses, the simple inspection of geometrical distances between regions and positions inside the eye band cannot indicate which regions correspond to the eyes. Hence, a more difficult approach based on region shape can be taken into account. However, in this paper we present a simple method to discriminate eye and hair region that performed with good results for a large number of sequences. In our approach the small hair regions inside the eye band are removed by analyzing the regions sizes in a larger window around the upper portion of the face. Inside this window, the hair corresponds to the region of large size.

Figure 5 shows the steps of the eye detection approach used in this paper. In the first stage, the pixel intensities inside the face regions are compared to a threshold θ and pixels with intensities lower than θ are extracted from the face region. The connectivity analysis of the extracted pixels generates connected sets of pixels, i.e. sets of pixels that are adjacent or touching. Each of these connected sets of pixels describe a low intensity region of the face.

In the next stage these regions are filtered with respect to the region size. Regions having a small number of pixels due to camera noise or shadows are removed. Large regions can not represent eyes and correspond in general to hair. The size of the regions selected at this stage is in the interval $[\theta_m, \theta_M]$ where θ_m is the minimum and θ_M is the maximum number of pixels allowed by our system to describe a valid eye region. Threshold values θ_m, θ_M are determined based on the size of the ellipse that characterizes the head region.

The remaining components are filtered based on the geometrical distances between eyes and the expected position of the eyes inside a rectangular window (eye band) centered in the ellipse centroid. The eye regions are determined by analyzing the minimum and maximum distance between the regions inside this band.

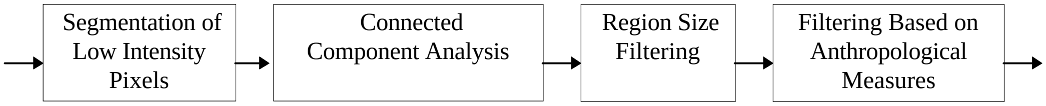
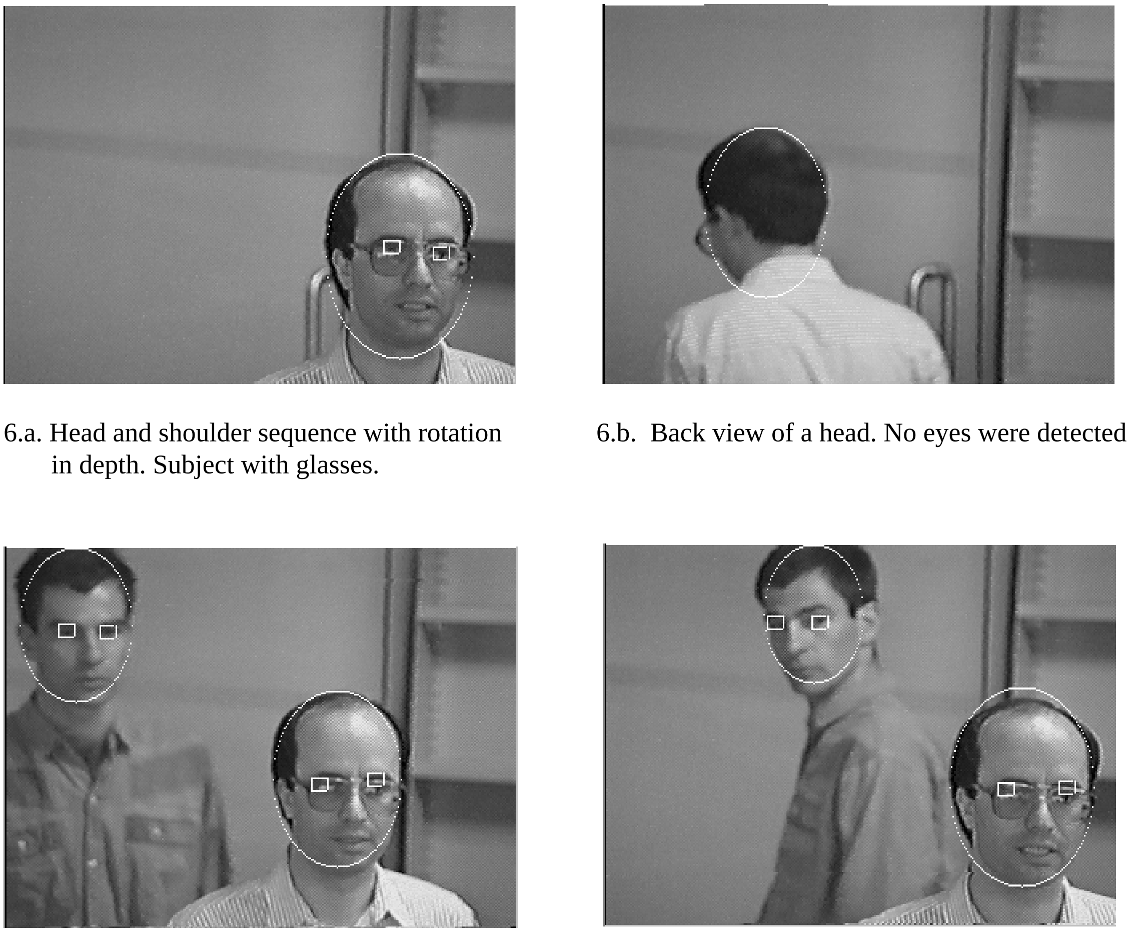

Segmentation of Low Intensity Pixels
Connected Component Analysis
Region Size Filtering
Filtering Based on Anthropological Measures

Figure 5. Eye Detection

7. RESULTS AND CONCLUSIONS

This paper presents a statistical model-based approach to human face detection. The system has been implemented and tested on a variety of different video sequences. Figure 6 shows the result obtained by running the system in a laboratory environment. This figure consists of four different scenarios generated to demonstrate the performances under different conditions such as non-frontal poses, multiple occluding people, back views, and faces with glasses. In Figure 6-(a) the face of a single person is detected. In this figure the ellipse is properly fitted around the face and the eyes are detected even with the glasses on the face. Figure 6-(b) shows the back view of a single person in the scene. In this figure, the ellipse is fitted around the head, but no eye is detected indicating the robustness of the eye detection module. Figure 6-(c) and 6-(d) show two scenarios in which two people are present in the scene. In both figures the body of one person is covering part of the body of the other person. In both cases the face and eyes are detected. The fact that the system is able to extract the faces and eyes in situations with body occlusion is a strong merit of the presented approach. In figure 6-(d) the face of the person in the back has a non-frontal position. Also due to different distances from the camera the size of the two faces are different. The faces of both persons are detected indicating the robustness of the system to variations in parameters such as size and position of the face.



6.a. Head and shoulder sequence with rotation in depth. Subject with glasses.

6.b. Back view of a head. No eyes were detected

6.c. Multiple people sequence in the presence of body occlusions.

6.d. Multiple people sequence in the presence of body occlusions and rotations in depth

Figure 6. Results for head and eye detection.

Future research is necessary to enhance the performance of the eye detection module and increase its robustness with respect to different face and hair colors. Currently we are continuing this research to extract the optimal (most frontal) pose of a person from a sequence of video frames in an uncontrolled environment.

8. REFERENCES

[1] Rama Chellapa, Charles L. Wilson and Saad Sirohey, “Human and Machine Recognition of Faces: A Survey”, in *Proceedings of the IEEE*, vol. 83, no. 5, pp 705-740, May 1995.

[2] Kah-Kay Sung , Tomasso Poggio, “Example-based Learning from View-based Human Face Detection”, in *Image Understanding Workshop*, pp 843-850, vol 2., 1994

[3] Matthew A. Turk, Alex Pentland, “Face Recognition Using Eigenfaces” in *Proceedings on International Conference on Pattern Recognition*, pp. 586-591, 1991.

[4] Christine Podilchuck, Xiaoyu Zhang, “Face Recognition Using DCT-Based Feature Vectors”, in *ICAASP*, pp 2144-2146, 1996.

[5] Ram R. Rao, Russell M. Meresereau, “On Merging Hidden Markov Models with Deformable Templates” in *Proceedings of International Conference on Image processing*, pp 556-559, vol. 3, 1995.

[6] A.L. Yuilee, P.W. Hallinan, and D.S. Cohen “Feature Extraction from faces using deformable templates”, in *International Journal of Computer Vision*, vol. 8, pp. 299-311, 1992.

[7] Luigi Stringa , “Eyes Detection for Face Recognition”, *Applied Artificial Intelligence*, vol.7, no. 4, pp 365-382, Oct-Dec 1993.

[8] Roberto Brunelli and Tomasso Poggio, “Face Recognition : Features Versus Templates”, *IEEE Transaction on Pattern Analysis and Machine Intelligence*, October 1993

[9] L. R. Rabiner, “A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition”, *Proceedings of the IEEE*, February 1989.

[10] Cristopher Wren, Ali Azerbayejani, Trevor Darrell, Alex Pentland, “Pfinder: Real Time Tracking of the Human Body”, *in SPIE Conference on Itegration Issues in large Commercial Media Delivery Systems*, October 1995.