1999 7th IEEE International Conference on
Emerging Technologies and Factory Automation
Proceedings
ETFA’99

October 18-21, 1999
UPC, Barcelona, Catalonia, Spain

Sponsored by
IEEE Industrial Electronics Society
IEEE Robotics and Automation Society
Universitat Politècnica de Catalunya

Edited by: J.M. Fuertes
A Wheelchair for Handicapped People by Face Tracking

Bergasa, M. Mazo, A. Gardel, J.C. García, A. Ortuño, A.E. Mendez

Department de Electrónica. Escuela Politécnica. Universidad de Alcalá
Campus Universitario s/n. 28805 Alcalá de Henares. MADRID. Spain
T:+34 91 885 6569-40 Fax: +34 91 885 6591
E-mail:bergasa@depeca.alcala.es http://www.depeca.alcala.es

This paper shows a guidance system for an electrical wheelchair for handicapped people by head and color face tracking system has been developed in order to compute head movements of the depending on them, some commands are generated to drive the wheelchair. The system is non-automatic initial setup, working even for people of different races. It is adaptive and, therefore, right and background changes in inside environments. It has been tested with several users and are given.

1. INTRODUCTION

Machine interaction which is very important the direction in which a person is facing information can be required for several automatic focus [1], teleconferencing with recognition [2], face identification in real-time, gaze driven panorama image viewer systems [4], lips readers [5], and mobility of disabled people [6], etc.

Department of University of Alcalá has more than 6 years, on artificial means of handicapped people. Nowadays, is being developed, within SIAMO system for Assisted Mobility) [7], in multi-functional wheelchair for disabled (Figure 1). This project includes an eye, by head movements, for cases of a present, a wheelchair prototype is a guidance method. A 3D simulator has been developed, to help the users in adapting system in a safer way.

The global system architecture and the methodology followed in its design. Experimental results are given and, conclusions about its performance have been drawn.

Figure 1. SIAMO Project

Applying the Kinematic model, linear and angular speed become angular speeds for every wheel ($\alpha_{cmd}$ and $\omega_{cmd}$) and they are sent to the low level control. In this level a PI controller has been designed to control the velocity of each wheel.

Next, the main parts of the system architecture will be explained.

2.1. Skin Segmentation

The UASGM method segments any person’s skin, even of different races, under changing light conditions and...
random backgrounds. To do this, a stochastic adaptive model of skin colors in a normalised RG (Red, Green) color space has been used.

The model is initialized by a clustering process. This divides the chromaticities of an image into a number of classes (k) between one and a maximum value \( k_{\text{max}} \). At each step, the k cluster centers are estimated using an approximate color histogram. After the centers are adjusted employing a competitive learning strategy (Vector Quantization algorithm [9]) in the closest center sense. Finally a clustering quality factor is computed for each topology. This factor for k classes is given by (1), where \( \text{tr}[\cdot] \) is the trace of a matrix and \( S_w \) and \( S_b \) denotes, respectively, the within-cluster and between-cluster scatter matrices.

\[
F_k = \text{tr}[S_w^{-1} S_b] \quad 1 \leq k \leq k_{\text{max}} \quad (1)
\]

\[
S_b = \frac{1}{k M} \sum_{i=1}^{k} M_i (m_i - m_0) (m_i - m_0)^T \quad (2)
\]

\[
S_w = \frac{1}{k M} \sum_{i=1}^{k} \frac{M_i}{M_{i+1}} \sum_{j=1}^{M_i} (x_{ij} - m_i)(x_{ij} - m_i)^T \quad (3)
\]

\[
m_0 = \frac{1}{M} \sum_{i=1}^{M} x_{ij} \quad ; \quad m_i = \frac{1}{M_i} \sum_{j=1}^{M_i} x_{ij} \quad (4)
\]

In these equations, \( k \) is the number of clusters, \( M_i \) is the number of pixels in the \( i \)th cluster, \( x_{ij} \) is a color pixel in the \( i \)th cluster, \( m_i \) is the mean of the \( i \)th cluster, \( m_0 \) is the mean of all of the feature vectors and \( M \) denotes the number of pixels to be clustered.

The process is repeated adding a new class at each step until the maximum number of classes \( k_{\text{max}} \) is reached. The maximum quality factor gives the number of classes that best fits the histogram distribution. Within each class the skin cluster is located using the distance between the center of the cluster and the skin color position. Then, the skin class is modeled as a Gaussian function \( N(m_i, C_i) \) and the parameters of the model are estimated by a linear combination of these using the maximum likelihood criterion.

The estimated mean vector, \( \hat{m}_S \), and the covariance matrix, \( \hat{C}_S \), will be calculated by:

\[
\hat{m}_S = \sum_{l=1}^{\nu} \alpha_l R_l
\]

\[
\hat{C}_S = \sum_{l=1}^{\nu} \beta_l R_l
\]

where, \( R_l \) are the previous mean vectors, \( \alpha_l \) is the previous covariance matrix; \( \alpha_l \) are the coefficients of the mean prediction and \( \beta_l \) are the coefficients of the covariance matrix prediction. To calculate the coefficients for the prediction, we have used the algorithm proposed by Anderson [10]. The basic idea of this algorithm is to iteratively estimate the coefficients \( \beta_l \) independently, where \( l \) denotes the \( l \)th iteration. The iteration...
2.2. Face Tracking

The use of the estimation theory for tracking in computer vision is well known [13]. On the skin blob some parameters are calculated to track the face: center of gravity (x,y), horizontal (h) and vertical (v) size of the face, being able to obtain the face position and orientation.

A zero-th order Kalman filter is used to estimate two independent state vectors: one of them for the horizontal variation (X_h = (x,h)) and the other one for the vertical variation (X_v = (y,v)). Two independent state vectors have been utilized because, in our application, users can do only horizontal and vertical head rotation movements. Then can be taken into account that horizontal and vertical movements are independent: horizontal size depends on the ‘x’ center position and vertical size on ‘y’ center. Horizontal and the vertical size of the face are calculated as two parameters, because the aspect ratio of the face can change with rotations. The state vectors (X_h, X_v) and their covariance matrices (C_{X_h}, C_{X_v}) are estimated in a recursive process composed by three phases: predict, match and update.

2.3. Commands generation

Estimated state vectors (\hat{X}_h, \hat{X}_v) and their derivatives (\dot{\hat{X}}_h, \dot{\hat{X}}_v) are inputs to a command generation state machine. Each state codifies one of the following commands: turn right, turn left, increase speed, decrease speed and no command. State transitions of the machine are achieved analysing the activation of some fuzzy conditions of input variables, based on thresholds. These are computed in the initial setup. Likewise, using the information given by eyes and mouth position (X_{eye}, X_{mouth}) the special commands on/off and forward/backward will be obtained. These positions are calculated by the analysis of the hollows on the skin blob and imposing some geometrical restrictions. Hollows appear in the blob because in the face there are some features like: eyes, mouth, eyebrows, etc, that have different colors related with the skin.

Figure 3 shows the actions that generates the commands. For that, we have followed a criterion of simplicity in the fulfillment of these actions and in the robustness of their detection. Therefore, if user turns the head to the right the wheelchair will turn to the same direction. This happens as well if he turns his head to the left. Head rising involves the increment of wheelchair speed, and when the user bows it, this will decrease the speed.

Every time he winks an eye the wheelchair changes state on/off, when lips are hidden changes again the command forward/backward. In this way the special actions for, at least, two seconds are related with
the 'special command' activation. Doing this, wrong commands are avoided as a consequence of normal blinks. On/off commands allows to user activate or deactivate the system by themselves, so when it is in the off state, it can make any kind of movement being secure on the fact that no other command will be activated.

because, working with the derivatives of the state the actions speed are taken into account. It is not to do one action, but it must be executed quickly, way some involuntary wrong actions are eliminated.

<table>
<thead>
<tr>
<th>I: Increase, D:Decrease, E:Equal</th>
<th>State Variables</th>
</tr>
</thead>
<tbody>
<tr>
<td>Actions with the head</td>
<td>h x y z</td>
</tr>
<tr>
<td>No Command</td>
<td>E E E E</td>
</tr>
<tr>
<td>Turn Right</td>
<td>I D D I</td>
</tr>
<tr>
<td>Turn Left</td>
<td>I I D I</td>
</tr>
<tr>
<td>Rising</td>
<td>E E I I D</td>
</tr>
<tr>
<td>Bows</td>
<td>E E D D I</td>
</tr>
</tbody>
</table>

Table 1. Fuzzy conditions in the evolution of the state variables for the different actions.

![Figure 3. Actions that generate the commands](image)

Table 1 presents the way in which the system recognizes the action that user has made, paying attention to some fuzzy conditions in the evolution of the state variables. Analysing the activation of these fuzzy conditions and their derivatives the commands are generated.

In figure 4 we can see an example of the right command activation for a sequence of right progressive rotations made by the user (30°, 60°, 90°). Here, the evolution of the state variables and their derivatives is shown.

Then, the values of the state variables for the "no command" position can be regarded as \((x_R, y_R, h_R, v_R)\). These last variables are used as well as reference positions in the calculation of the fuzzy conditions to do the state transitions.

As the system is able to detect all the movements, the 30° one, in the 250 samples, as the movement was made very slowly. This happens

2.4. High level Control

Commands are sent to another state machine which implements high level control and will generate the

108
command to the wheelchair \((V_{cmd})\) of time. It works as it’s shown in the state machine: it has six states: "Stop", "No action", "Decelerate", "Turn right" and "Forward", "Backward", "Left". At a predetermined initial value a final value, positive or negative, is reached. Then "Increase speed" and "Decrease speed" commands increase or decrease linear motion of time and depending on the direction (forward/backward), up to certain preset levels.

The command on/off allows to stop the process.

The system uses a visual feedback loop, which detects according to the current instance, if the system detects a right wheelchair will turn to the right until it is on board the wheelchair presents shown in figure 6 [14]. A LonWorks network has been developed, where nodes can easily be added because of its design.

The wheelchair prototype is also provided with an ultrasonic ring to increase safety during navigation and a user interface composed by a joystick and a LCD.

4. EXPERIMENTAL RESULTS

The vision system is able to process up to 10 images per second, with a resolution of 128x128 pixels. During the testing stage it great robustness exhibited in the velocity commands. However, some details are still to be fixed on the special commands. In order to increase the system controllability two switches have been included. One of them is used to activate the on/off command, while the second is intended to activate the forward/backward command. 10 commands per second are issued to the low level controller. The maximum wheelchair velocity was set to 1 m/s.

The system was tested for 5 different users in the Electronics Department’s lab after some training on the simulator. In figure 7 we show the image of one of the users during a test which was performed by that user during one of the test sessions in figure 8. The evolution of the maximum wheelchairs' velocities is presented and test lasted 100
seconds, taking 5 samples per second. In this figure the accelerations of the wheelchair during straight sections can be clearly appreciated. Also, the wheelchair decreases its speed before performing a curve. In this case, six left turns were executed, yielding great angular velocity peaks in the figure.

5. CONCLUSIONS AND FUTURE WORK

The conclusions obtained by the users after performing the test on the navigation system are presented below:

• It is non-intrusive, as it is passive, and there is no need for additional elements.

• Guidance complexity is decreased as more training is performed. We must also take into consideration that the camera is 80 cms in front of the user and, therefore, the system requires certain space for safe manoeuvring.

• The simple commands set and the wheelchair response allow for easy controllability in environments with not too many obstacles.

• Audible feedback is included to ensure proper command acknowledgement.

• It works well in indoor environments, where suitable illumination can be provided, decreasing the performance as light conditions get poorer. In outdoor environments there is no uniform illumination (shadows, direct sun light, etc), decreasing also the system capabilities.

Some future guidelines are to embed the system on a hardware platform and to develop a 3D model of the head in order to robustly obtain the gaze direction (not depending on the particular user) and perform the wheelchair control according to it.

6. ACKNOWLEDGEMENTS

This work has been financed by the CICYT (Spanish Interministerial Science and Technology Commis through the SIAMO project TER96-1957-C03-01.
7. REFERENCES


