Robotica: page 1 of 18. © Cambridge University Press 2011
doi:10.1017/S0263574711000919

Robotic wheelchair controlled through a vision-based interface

Elisa Perez†,*, Carlos Soria‡, Oscar Nasisi‡, Teodiano Freire Bastos§ and Vicente Mut‡

†*Gabinete de Tecnología Médica, Facultad de Ingeniería, Universidad Nacional de San Juan, Argentina*
‡*Instituto de Automática, Facultad de Ingeniería, Universidad Nacional de San Juan, Argentina*
§*Centro Tecnológico, Universidade Federal do Espírito Santo, Brazil*

(Accepted July 7, 2011)

SUMMARY
In this work, a vision-based control interface for commanding a robotic wheelchair is presented. The interface estimates the orientation angles of the user's head and it translates these parameters in command of maneuvers for different devices. The performance of the proposed interface is evaluated both in static experiments as well as when it is applied in commanding the robotic wheelchair. The interface calculates the orientation angles and it translates the parameters as the reference inputs to the robotic wheelchair. Control architecture based on the dynamic model of the wheelchair is implemented in order to achieve safety navigation. Experimental results of the interface performance and the wheelchair navigation are presented.

KEYWORDS: Assistive robots; Service robots; Computer vision; Mobile robots; Navigation.

1. Introduction

Rehabilitation robotics have evolved[1–4] in the last decades due to different factors: technological evolution, demands by people with disabilities, and the growth in world population of people with disabilities. This has seen changes to medical model of rehabilitation, which now seeks to provide autonomy, freedom, and improved quality of life for severely disabled people. Therefore, robotic wheelchairs are extremely important for assistance and service to people with special needs.

People who are severely disabled, such as paraplegics, often have very special needs and require specific and complex devices to provide them with autonomy and mobility. Suitably equipped robotic wheelchairs[5–7] can fulfil the need of this sector of the society. While the mechanical aspects of these chairs have been extensively developed, there is still work required to develop control interfaces and mechanisms that can support the people with special needs to use these chairs effectively.

A Number of researchers have developed and reported different modes of control for robotic wheelchairs. In ref. [8], an agent-based autonomous robotic wheelchair controller is used to command an autonomous robotic wheelchair in indoor environments. The navigation system of the autonomous robotic wheelchair is developed as a group of intelligent agents, including the robotic wheelchairs path planning and the fuzzy-logic-based motion control and obstacle avoidance. While an excellent approach, this paper only presents simulations results and not the experimental results. In ref. [9], a robotic wheelchair adapted to cognitive disabled children is presented. Three human machine interfaces were provided: speech recognition based, motion interpreter, and visual feedback. The results are very promising, since the first time the children used the wheelchair they managed to drive it. However, the disadvantage of this work is that it does not provide an optimized control system that would ensure the velocity of the wheelchair. Further, such a system is not suitable for patients with severe brain and spinal injury who do not have recognizable speech and haptic sensing abilities. In ref. [10], the design and implementation of a hands-free system for intelligent wheelchair is described. A head-gesture-based interface is developed, based on the organic integration of the Adaboost face detection algorithm and Camshift object tracking algorithm. The control architecture has two control modes, namely, intelligent control and manual control. This paper does not show a control law that guaranties that the wheelchair would reach the desired velocities. In the work,[11] the dynamic multivariable model of the wheelchair system is obtained. The control structure developed combining neural network controllers and decoupling techniques. This control structure provides solutions to nonlinear multivariable control problems. This work does not present real experimental results; the authors only present system simulations. In ref. [12], it is described the concept of the collaborative wheelchair assistant (CWA). The CWA is to rely on the user's motion planning skills while the maneuvering with flexible path guidance. The user decides where to go and controls the speed, while the system guides the wheelchair along software defined guide paths.

In this paper, a novel assistive robotic system for people with severely motor disabilities is presented. The proposal includes a vision-based interface (VBI) as well as the control algorithms in order to allow motor disable people to command a robotic wheelchair. The VBI detects and tracks the movements of the head of a person and traduces these movements into reference signals for the control algorithms. In the proposed VBI, two image processing techniques work in parallel to estimate the position parameters. Then, a fusion

* Corresponding author. E-mail: eperez@gateme.unsj.edu.ar

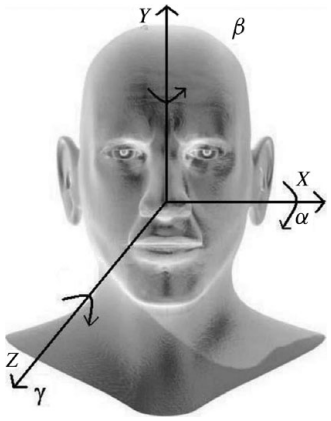

Fig. 1. Rotation angles of the head in 3D space.

process of the outcomes of these two techniques provides the references signals for the control laws. The fusion process is implemented in order to improve accuracy in the estimation of the head's position obtaining a better performance of the human–machine interface. It is important to note that, different from other works found in the literature,[9,10,13,14] our approach includes a control algorithm based on the dynamic model of the wheelchair in order to regulate its velocities, achieving a more safety navigation.[15] The proposed VBI is evaluated in static experiments obtaining performances comparable with the best techniques reported in the literature.[16] Additionally, the performance of the VBI is tested in a robotic wheelchair control showing the experimental results.

In the robotic wheelchair developed earlier by the authors,[17] a machine–human interface is used allowing the user to control the robotic wheelchair by blinking the eyes, eye movements, and using brain signals recorded from the surface. The navigation system of this wheelchair allows the user (or caregiver) different driving paradigms: user controlled point-to-point, autoguide, and a hybrid approach based on both point-to-point and autoguide controller. In the autoguide mode, the wheelchair tracks a predefined metal pathway using magnetic sensors. Therefore, the proposed interface further enhances previous work by providing the user the ability to command the wheelchair with head movements.

This work is organized in the following way: Section 2 describes the VBI, Section 3 describes the design the proposed controller, the experimental results are given in Section 4, and Section 5 gives the conclusions.

2. Vision-Based Interface

The interface reported in this paper obtains two parameters from the video data of the user head and generates two independent command signals to navigate the wheelchair. The parameters used are the orientation angles of head (α and β angles) in the space. These are with respect to axes X and Y (see Fig. 1). The research and development report consists of the implementation and fusion of the angle β obtained by two image processing algorithms. Angle β is used to generate angular velocity to the wheelchair and the angle

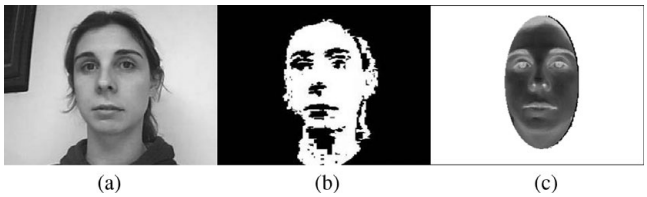

Fig. 2. (a) Original image. (b) Skin detection. (c) Luminance image with ellipse.

α to determine the linear velocity. The image processing techniques will be described next.

2.1. Detection and tracking of facial features for estimating the pose of the head of the person

The first task in this method is the detection of the face. This determines if the face is in the scene.[18] One technique for face detection is based on features, which extracts image features and makes a tracking of their movements from an image to another one. The more common the image features are regions, skin, contours, and landmarks.[19,20] In this work, the detection of skin is chosen for detecting and tracking the face.

One general shortcoming in skin-color-based detection is the impact of lighting conditions. In refs. [21–23], different preprocessing methods of the images for illumination compensation are proposed. To mitigate this shortcoming, light compensation technique was employed. For this purpose, histogram equalization of the input image was used. This method is easier and faster to implement, different from previous techniques that require high computing cost. Furthermore, it improves the quality of the image by increasing the dynamic range of the pixels and enhancing the image contrast.

The second step is the detection of the skin by segmentation in the color space YCbCr. The original image is transformed from RGB to YCbCr space by:[24]

$$\begin{aligned} Y &= 0.299R + 0.587G + 0.114B, \\ Cb &= -0.169R - 0.332G + 0.500B, \\ Cr &= 0.500R - 0.419G - 0.018B, \end{aligned} \tag{1}$$

where Y, Cb, and Cr represent the value of a pixel in each channel of the color space YCbCr and R, G, and B represent the value of the same pixel in each channel of the color space RGB. The YCbCr space is chosen because its components have less variation to different skin tones.[25] Therefore, this space is more flexible and stable for the system. Segmentation was performed by thresholding of Cr and Cb components[26] in the YCbCr color space. The threshold values were empirical determined and were

$$\begin{aligned} 130 < Cr < 170, \\ 70 < Cb < 127. \end{aligned}$$

The third step is to select the maximum connected zone corresponding to the skin Fig. 2(b). The background objects were eliminated.

The fourth step is to describe an ellipse using the image moments described by ref. [27], which were used to identify

the centroid, mayor axis L1, and minor axis L2. This ellipse defines the region of interest (ROI) for extracting facial features in the luminance image (Fig. 2c).

The fifth step is the extraction and tracking of the facial features. Among the different facial features, the eyes and the mouth are the most stable features to estimate the 3D pose of the head. There are many different approaches to locate facial features,[28–30] each approach having its own limitations. One of the shortcomings of image-based command systems is the poor reliability. To improve the performance, integration of two methods for facial features extraction and tracking is proposed in this paper. This consists of (a) *K*-means algorithm and (b) normalized correlation. *K*-means technique is used to determine the centroid of a set of features points[31] while normalized correlation is used to identify the eyes' region based on the template of the eyes. The algorithm proposed can be subdivided into three well-defined steps: Step (1) extraction of features points and computation of the centroid of the facial features in the image; Step (2) normalized correlation; Step (3) the results obtained in the previous steps are combined.

Step (1): The facial features are searched in the image obtained after segmentation (Fig. 2c). The algorithm of the feature points is used to detect the eyes and mouth. This method is used to localize the eyes and mouth because these facial features are usually brighter than the surrounding.[32] The choice of the features points do not belong to any geometric organization observed in the image of the face. The selected features are contours or corners.

The next step is the identification of the corner points in the contour, by using a method based on the spatial gradient of the image.[32] With this aim, let consider an image point p, a region $\mathbf{Q}$ (of 5×5 pixels in this work) surrounding it, and a matrix $\mathbf{C}_p$ defined as

$$\mathbf{C}_p = \begin{bmatrix} \sum\limits_{\mathbf{Q}} E_{ix}^2 & \sum\limits_{\mathbf{Q}} E_{ix}E_{iy} \\ \sum\limits_{\mathbf{Q}} E_{ix}E_{iy} & \sum\limits_{\mathbf{Q}} E_{iy}^2 \end{bmatrix}, \tag{2}$$

where E_{ix} and E_{iy} are the components of the gradient for each point in the region $\mathbf{Q}$. As $\mathbf{C}_p$ is symmetric, it can be diagonalized in the following as:

$$\mathbf{C}_p = \begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix}, \tag{3}$$

where λ_1 and λ_2 are the eigenvalues of the matrix $\mathbf{C}_p$.

A corner is a result of two contours with high gradients. Therefore, the eigenvalues of matrix $\mathbf{C}_p$ will both have a high value in the neighborhood of a corner different from the neighborhood of an edge, where matrix $\mathbf{C}_p$ will present one eigenvalue with a high value and the other one with a low value. Then, through the geometric interpretation of the eigenvalues of $\mathbf{C}_p$, the point p will represent a corner if the minor eigenvalue is large enough. That is, if the minor eigenvalue is higher than some defined threshold (in this work, this threshold was selected as 100) then the image point p represents a corner.

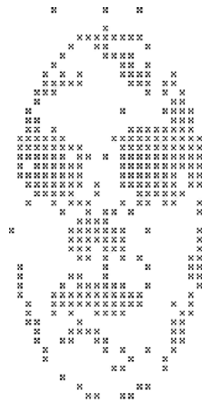

Fig. 3. Image with feature points.

(a) (b)

Fig. 4. (a) Image obtained using correlation. (b) Template of the eye.

The selection of this threshold and the size of region $\mathbf{Q}$ depends on the image considered. Typical size for $\mathbf{Q}$ varies between 3×3 and 11×11 pixels. On the other hand, the threshold for the minimum eigenvalue can be adopted by making an off-line analysis of its histogram in a typical image.

Appling this method to each pixel of the elliptical ROI, a set of features points that contain strong contours (eyes and mouth) is obtained (Fig. 3).

Once the feature points are determined, the centroids of the two regions of interest (eyes) are obtained. This is achieved by clustering the feature points and discarding points that were not associated with regions of the facial features using *K*-mean, which is simple and efficient.

Step (2): The second step is the optimization of feature extraction based on the correlation. For this purpose, a subimage of size 50×36 pixels is used. This subimage from a known image of the frontal face (Fig. 4a) is compared with the eyes region on the luminance image using fast correlation to identify the eye location (Fig. 4b).

Step (3): The average centroid values of the regions associated to each eye obtained in the previous technique are introduce in a Kalman filter.[33] This filter considers a kinematics model of first order, i.e., considering constant velocity[32] whose states correspond to the measurement of the centroids and its velocities. The values of the covariance matrix of the Kalman filter $\mathbf{Q}_{\text{kalman}}$ and $\mathbf{R}_{\text{kalman}}$ are $\mathbf{I}_{2\times2}$ and $5 \times \mathbf{I}_{2\times2}$, respectively. These values were adjusted offline using previous centroids data measurements in order to obtain a good trade-off between filtering performance and features tracking. If the feature position is not within the uncertainty limits derived from the error covariance matrix or feature is lost, then the predicted point features are used instead of the measurement. In Fig. 5, the facials features obtained are shown.

It is important to note that the user is allowed to close his eyes and the VBI will continue detecting them since the proposed system detects the corners of the regions of the eyes instead of the pupils or the ocular globes.

Fig. 5. Image with the centroids of facial features estimated.

2.1.1. Estimation of the orientation and position of head. The variation of the head's pose can be assumed to follow the pattern of the rigid movement, and the image of the face can be considered by different projections on the 2D image plane. Based on this assumption, homographic transformation is used to obtain the rotation angle of the head by using the projection of the centroids of the eyes onto the image plane.

Homographic transformation is the relationship between the projected coordinates on the image plane when one plane in the space moves with rigid movement, from instant t to t'. The eyes are assumed to be placed on the same plane in the space.

The displacement 3D of the rigid object in Cartesian coordinates is calculated through the following equation:

$$\mathbf{X}' = \mathbf{R}\mathbf{X} + \mathbf{T}, \tag{4}$$

where $\mathbf{R}$ is the 3×3 rotation matrix, $\mathbf{T}$ is a 3×1 translation vector, $\mathbf{X}$ and $\mathbf{X}'$ are coordinates at the times t and t', respectively, with respect to the rotation center. The rotation matrix $\mathbf{R}$ is expressed as a function of the angles α, β, and γ, which are the orientation angles of the head in the space (see Fig. 1)

$$\mathbf{R} = \mathbf{R}_\gamma \mathbf{R}_\beta \mathbf{R}_\alpha, \tag{5}$$

with

$$\mathbf{R}_\gamma = \begin{bmatrix} \cos\gamma & -\sin\gamma & 0 \\ \sin\gamma & \cos\gamma & 0 \\ 0 & 0 & 1 \end{bmatrix}; \quad \mathbf{R}_\beta = \begin{bmatrix} \cos\beta & 0 & -\sin\beta \\ 0 & 1 & 0 \\ \sin\beta & 0 & \cos\beta \end{bmatrix};$$

$$\mathbf{R}_\alpha = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\alpha & -\sin\alpha \\ 0 & \sin\alpha & \cos\alpha \end{bmatrix}.$$

Since the features points are in the same plane in the 3D space and all of them have the z-coordinate equal to zero, the Eq. (4) can be written as

$$\mathbf{X}' = \mathbf{H}\mathbf{X}, \tag{6}$$

where the matrix $\mathbf{H}$ represents the homography transformation, which is a function of the matrix $\mathbf{R}$ and the translation vector $\mathbf{T}$

$$\mathbf{H} = [\mathbf{R} \mid \mathbf{T}] = \begin{bmatrix} h_1 & h_2 & h_3 & h_4 \\ h_5 & h_6 & h_7 & h_8 \\ h_9 & h_{10} & h_{11} & h_{12} \end{bmatrix}. \tag{7}$$

This homographic matrix posses 11 unknown parameters; therefore, six points are needed to find the rotation and translation parameters. In this way, the rotation parameters values correspond with the orientation angles α, β_p (being β_p the β-angle obtained by the homographic transformation), and γ of the head and the translation parameter corresponds with the distance between the head and the camera. In this research only, the angle β is used. Then, the homographic matrix has only four parameters unknowns, and it can be solved with only two point's projections on the image plane.

The centroids of the eyes are used to show the homographic transformation. The matrix $\mathbf{H}_\beta$ that transforms the points in the space on the image plane can be expressed in homogeneous coordinates as follows:

$$\mathbf{H}_\beta = \mathbf{H}_1\mathbf{H}_2 = \begin{bmatrix} \mathbf{R}_\varphi & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \mathbf{R}_{\beta p} & \mathbf{T} \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} \mathbf{R}_\varphi \mathbf{R}_{\beta p} & \mathbf{R}_\varphi \mathbf{T} \\ 0 & 1 \end{bmatrix}, \tag{8}$$

where $\mathbf{T}$ is $\mathbf{T} = [0\ 0\ d]^{\mathrm{T}}$, $\mathbf{R}_{\beta p}$ is the rotation matrix when the head has rotated respect to axis Y and $\mathbf{R}_\varphi$ the camera orientation matrix with respect to axis Y, which is described in the following way:

$$\mathbf{R}_\varphi = \begin{bmatrix} \cos\varphi & 0 & -\sin\varphi \\ 0 & 1 & 0 \\ \sin\varphi & 0 & \cos\varphi \end{bmatrix}. \tag{9}$$

Finally, the $\mathbf{H}_\beta$ transformation matrix in homogeneous coordinates is

$$\mathbf{H}_\beta = \begin{bmatrix} \cos(\beta_p + \varphi) & 0 & -\sin(\beta_p + \varphi) & -d\sin\varphi \\ 0 & 1 & 0 & 0 \\ \sin(\beta_p + \varphi) & 0 & \cos(\beta_p + \varphi) & d\cos\varphi \\ 0 & 0 & 0 & 1 \end{bmatrix}. \tag{10}$$

To obtain the coordinates of the point relative to the coordinate system associated to the camera, the matrix $\mathbf{H}_\beta$ defined in Eq. (10) is used. The coordinates of the points are

$$\begin{bmatrix} X_{\mathrm{cam}} \\ Y_{\mathrm{cam}} \\ Z_{\mathrm{cam}} \\ 1 \end{bmatrix} = \begin{bmatrix} \cos(\beta_p + \varphi) & 0 & -\sin(\beta_p + \varphi) & -d\sin\varphi \\ 0 & 1 & 0 & 0 \\ \sin(\beta_p + \varphi) & 0 & \cos(\beta_p + \varphi) & d\cos\varphi \\ 0 & 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}. \tag{11}$$

The model of perspective transformation (pinhole model) of the camera is considered and the coordinates of the point on the image plane are

$$\mathbf{x} = \mathbf{K}\mathbf{R}_\varphi[\mathbf{R}_{\beta p} \mid \mathbf{T}]\mathbf{X}. \tag{12}$$

In Eq. (12), **X** is the vector of coordinates of the point in the world; **x** is the vector of coordinates of the point on the image plane; $\mathbf{R}_{\beta p}$ and **T** are matrices that contain information of the camera position and orientation respect to the coordinate system of the real 3D space. These matrices are denominated external parameters, and **K** is the matrix that contains information of the intrinsic parameters of the camera is expressed as

$$\mathbf{K} = \begin{bmatrix} \delta_x & 0 & x_0 & 0 \\ 0 & \delta_y & y_0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}, \tag{13}$$

where δ_x and δ_y represent the focal distance of the camera, which are expressed in pixels; x_0 and y_0 are the coordinates of the center of the image. Then, the lateral coordinates on the image plane (expressed in pixels) are

$$\begin{bmatrix} \chi_{1i} \\ \chi_{2i} \\ \chi_{3i} \end{bmatrix} = \begin{bmatrix} \delta_x & 0 & x_0 & 0 \\ 0 & \delta_y & y_0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} X_{\text{cam}} \\ Y_{\text{cam}} \\ Z_{\text{cam}} \\ 1 \end{bmatrix}. \tag{14}$$

Therefore, from Eqs. (11) and (14), the coordinates of the point projected on the image plane are obtained as follows:

$$\begin{bmatrix} \chi_{1i} \\ \chi_{2i} \\ \chi_{3i} \end{bmatrix} = \begin{bmatrix} \delta_x & 0 & x_0 & 0 \\ 0 & \delta_y & y_0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \times \begin{bmatrix} \cos(\beta_p + \varphi) & 0 & -\sin(\beta_p + \varphi) & -\sin\varphi \\ 0 & 1 & 0 & 0 \\ \sin(\beta_p + \varphi) & 0 & \cos(\beta_p + \varphi) & d\cos\varphi \\ 0 & 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}. \tag{15}$$

The above Eq. (15) can be split into three equations:

$$\begin{aligned} \chi_{1i} &= \delta_x[X\cos(\beta_p + \varphi) - Z\sin(\beta_p + \varphi) - d\sin\varphi] \\ &\quad + x_0[X\sin(\beta_p + \varphi) + Z\cos(\beta_p + \varphi) + d\cos\varphi], \\ \chi_{2i} &= \delta_y Y + y_0[X\sin(\beta_p + \varphi) + Z\cos(\beta_p + \varphi) + d\cos\varphi], \\ \chi_{3i} &= [X\sin(\beta_p + \varphi) + Z\cos(\beta_p + \varphi) + d\cos\varphi], \end{aligned} \tag{16}$$

whereas the positive direction of the *Y*-axis is down, the sign of the second equation of Eq. (16) must be changed. Furthermore, the system equations (16) must be expressed in homogeneous coordinates and the following system of equation is obtained:

$$x_i = \frac{\delta_x[X\cos(\beta_p + \varphi) - Z\sin(\beta_p + \varphi) - d\sin\varphi]}{[X\sin(\beta_p + \varphi) + Z\cos(\beta_p + \varphi) + d\cos\varphi]} + x_0, \tag{17}$$

$$y_i = \frac{-\delta_y Y}{[X\sin(\beta_p + \varphi) + Z\cos(\beta_p + \varphi) + d\cos\varphi]} + y_0. \tag{18}$$

The coordinate *Z* is equal to zero since the eyes are considered in the same plane. Therefore, the Eqs. (17) and (18) can be arranged as follows:

$$\begin{aligned} &X\sin(\beta_p + \varphi)(x_i - x_0) + d\cos\varphi(x_i - x_0) \\ &\quad -\delta_x[X\cos(\beta_p + \varphi) - d\sin\varphi] = 0, \\ &X\sin(\beta_p + \varphi)(y_i - y_0) + d\cos\varphi(y_i - y_0) + \delta_y Y = 0. \end{aligned} \tag{19}$$

If two points are considered in the space (right eye and left eye in the present work) and the projections' points are on the image plane, a four equations system with four unknown variables is obtained. The unknown variables are functions of the parameters: *d* (distance between the face and camera), and the angles φ and β_p. The system of equation obtained is

$$\begin{bmatrix} X_1(x_{1i} - x_0) & -\delta_x X_1 & (x_{1i} - x_0) & \delta_x \\ X_1(y_{1i} - y_0) & 0 & (y_{1i} - y_0) & 0 \\ X_2(x_{2i} - x_0) & -\delta_x X_2 & (x_{2i} - x_0) & \delta_x \\ X_2(y_{2i} - y_0) & 0 & (y_{2i} - y_0) & 0 \end{bmatrix} \begin{bmatrix} \sin(\beta_p + \varphi) \\ \cos(\beta_p + \varphi) \\ d\cos\varphi \\ d\sin\varphi \end{bmatrix} = \begin{bmatrix} 0 \\ -\delta_y Y_1 \\ 0 \\ -\delta_y Y_2 \end{bmatrix}, \tag{20}$$

where (X_1, Y_1) are coordinates of the right eye and (X_2, Y_2) are coordinates of the left eye. (x_{1i}, y_{1i}) and (x_{2i}, y_{2i}) are the coordinates of the centroids of the right and left eyes, respectively, in the image plane. In this work, physical measurements (X_i, Y_i) are obtained off-line as the mean value of the eyes positions, considering a set of 20 persons.

The solution of Eq. (20) is

$$\begin{aligned} &\sin(\beta_p + \varphi) = \delta_y \frac{Y_2(y_{1i} - y_0) - Y_1(y_{2i} - y_0)}{(y_{1i} - y_0)(y_{2i} - y_0)(X_1 - X_2)}, \\ &\cos(\beta_p + \varphi) = \\ &-\frac{\delta_y}{\delta_x} \frac{Y_1(y_{2i} - y_0)(x_{1i} - x_0) - Y_2(y_{1i} - y_0)(x_{2i} - x_0)}{(y_{1i} - y_0)(y_{2i} - y_0)(X_1 - X_2)}, \\ &d\cos\varphi = \frac{\delta_y[Y_1 X_2(y_{2i} - y_0) - Y_2 X_1(y_{1i} - y_0)]}{(y_{1i} - y_0)(y_{2i} - y_0)(X_1 - X_2)}, \\ &d\sin\varphi = \\ &\frac{\delta_y}{\delta_x} \frac{[Y_2 X_1(y_{1i} - y_0)(x_{1i} - x_0) - Y_1 X_2(y_{2i} - y_0)(x_{1i} - x_0)]}{(y_{1i} - y_0)(y_{2i} - y_0)(X_1 - X_2)}. \end{aligned} \tag{21}$$

Thus, the angle of head orientation, β_p and parameters φ and *d* were obtained. The parameter β_p is used in the signal fusion step.

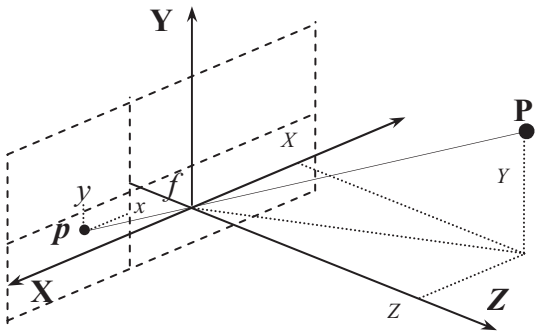

Fig. 6. Coordinate system.

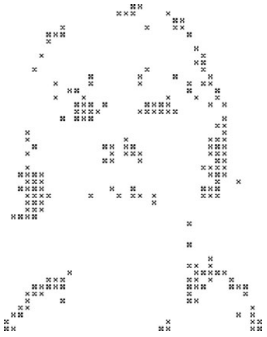

Fig. 7. Feature points.

Fig. 8. Estimated optical flow.

2.2. Estimation of the head movement by optical flow

The projection of the 3D movement of a rigid object forms a movement field on the image. This is the projection of the 3D speed field onto the image plane and it is the 2D temporal function for the relative spatial variation between the view camera and the observed scene.[32] The equations of the movement field can be obtained by projecting the mobile point in the 3D space to the image plane. For this projection, the origin is located at the optical center of the camera and *X*- and *Y*-axes are defined in such a way that they make a basis for the image plane. The *Z*-axis is the same as the optical axis of the camera. This situation is described in Fig. 6.

Let $\mathbf{P} = [X\ Y\ Z]^{\mathrm{T}}$ be a point in the 3D space. The relative movement between the point $\mathbf{P}$ in the world space and the camera can be described by

$$\mathbf{V} = \frac{\mathrm{d}\mathbf{P}}{\mathrm{d}t} = -\mathbf{T} - \boldsymbol{\omega}_{\mathbf{f}} \times \mathbf{P}, \tag{22}$$

where $\mathbf{T} = [T_x\ T_y\ T_z]^{\mathrm{T}}$ is the translational component, and $\boldsymbol{\omega}_{\mathbf{f}} = [\omega_x\ \omega_y\ \omega_z]^{\mathrm{T}}$ is the rotational component. Since the movement corresponds to a rigid object, $\mathbf{T}$ and $\boldsymbol{\omega}_{\mathbf{f}}$ are the same for any point $\mathbf{P}$.

Thus, the movement field $\mathbf{v} = [u_f\ v_f]^{\mathrm{T}}$ in the image plane can be calculated by

$$\begin{aligned} u_f &= \frac{f}{Z^2}(ZV_x - XV_z), \\ v_f &= \frac{f}{Z^2}(ZV_y - YV_z), \end{aligned} \tag{23}$$

where V_x, V_y, V_z being the components of $\mathbf{V}$ defined in Eq. (22). Introducing Eq. (22) into Eq. (23), the following equation, expressed in matrix form, is obtained:

$$\begin{bmatrix} u_f(x,y) \\ v_f(x,y) \end{bmatrix} = \frac{1}{Z}\begin{bmatrix} -f & 0 & x \\ 0 & -f & y \end{bmatrix}\begin{bmatrix} T_x \\ T_y \\ T_z \end{bmatrix} + \begin{bmatrix} \frac{xy}{f} & -f - \frac{x^2}{f} & y \\ f + \frac{y^2}{f} & -\frac{xy}{f} & -x \end{bmatrix}\begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix}. \tag{24}$$

In Eq. (24), two terms can be distinguished: the first one describes the component of the movement field of the image due to the translation of the object, which depends on the depth inverse in each point. The second term is the component of the movement field of the image due to the rotation of the object, which is independent of the depth. Once obtained the relation between the 3D space and its projection on the image plane, the concept of optical flow follows.

The optical flow or image speed is the bidimensional field of apparent speed in the image plane associated with the variation of patterns of brightness intensity in an image. The field can be produced due to the movement of the observer, the movement of the objects in the scene or the apparent movements. The values of optical flow u_f and v_f are calculated through the relationship between the variation in the brightness intensity of the image and the field movement, assuming that it remains constant under the movement.

The optical flow is estimated in the points that belong to strong contour in the face, which are calculated as explained in Section 2.1. Figure 7 shows the points calculated. In this work, the optical flow is calculated by implementing the method of ref. [34]. This method uses the Kalman filter to accomplish a data fusion step, obtaining a robust optical flow. The optical flow calculated is shown in Fig. 8. The estimation is carried out between two successive frames, I_t and I_{t+1}.

2.2.1. Estimation of the head movement. An estimation of the 3D movement is obtained by calculating the parameters of the successive rotation and translation between two frames, using the movement field in 2D. In the previous sections, the steps for obtaining the optical flow were explained. In this section, it will be explained how the parameters of the head pose are obtained from the optical flow.

The projection of the movement in 3D in the movement field 2D is given by Eq. (26). It is assumed that there is a movement of the head in the image (and all the background remains static), and the movement of the rigid body between two frames is infinitesimal. Therefore, the angular velocities

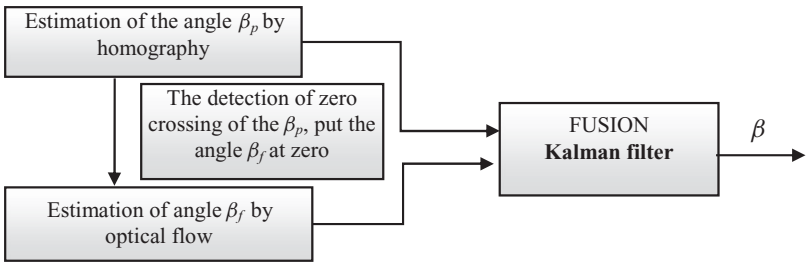

Fig. 9. Diagram on blocks of the fusion.

can be directly replaced by the values of the angles. The variation of the angles cannot be higher than 5°.[35] This assumption does not represent an important restriction since the webcam used in this work needs only 100 ms to process two frames, which is a very short time to make big head movements.

Equation (22) can be rewriting as follows:

$$\begin{aligned} V_x &= -T_x - \beta_f Z + \gamma_f Y, \\ V_y &= -T_y - \gamma_f X + \alpha_f Z, \\ V_z &= -T_z - \alpha_f Y + \beta_f X, \end{aligned} \tag{25}$$

where α_f is the rotation angle respect to X-axis, β_f is the rotation angle respect to Y-axis, and γ_f is the rotation angle respect to Z-axis. Subscript f refers to the variables obtained by the optical flow technique.

If $t_x = \frac{T_x}{Z}$, $t_y = \frac{T_y}{Z}$, and $t_z = \frac{T_z}{Z}$ and reorganizing Eq. (24), the following equation system is obtained:

$$\begin{bmatrix} u_{fi} \\ v_{fi} \end{bmatrix} = \begin{bmatrix} -f & 0 & x_i & \frac{x_i y_i}{f} & -\frac{f^2 + x_i^2}{f} & y_i \\ 0 & -f & y_i & \frac{f^2 + y_i^2}{f} & -\frac{x_i y_i}{f} & -x_i \end{bmatrix} \times \begin{bmatrix} t_x \\ t_y \\ t_z \\ \alpha_f \\ \beta_f \\ \gamma_f \end{bmatrix}. \tag{26}$$

The system of Eq. (26) is solved using the recursive least mean square, minimizing the error

$$||\mathbf{A}\mathbf{\Phi} - \mathbf{b}|| = E, \tag{27}$$

where

$$\mathbf{A} = \begin{bmatrix} -f & 0 & x_i & \frac{x_i y_i}{f} & -\frac{f^2 + x_i^2}{f} & y_i \\ 0 & -f & y_i & \frac{f^2 + y_i^2}{f} & -\frac{x_i y_i}{f} & -x_i \end{bmatrix},$$

$$\mathbf{b} = \begin{bmatrix} u_{fi} \\ v_{fi} \end{bmatrix},$$

$$\mathbf{\Phi} = [t_x \quad t_y \quad t_z \quad \alpha_f \quad \beta_f \quad \gamma_f]^{\mathrm{T}}.$$

This way, the three orientation angles of the head in the 3D space are obtained. However, only the angles β_f and α_f are used to command the robotic wheelchair.

A fusion process between the angles β_f and β_p will be later carried out, and the angle α_f will be used in the controller without any fusion process. In the following section, a brief description of the fusion between angle β_f and the angle β_p is presented.

2.3. Fusion

The angular values β_p and β_f, obtained by both techniques, are introduced in a Kalman Filter, in which the angles fusion is carried out (see Fig. 9). The fusion is a technique that can be applied when there is redundant information. The fusion decreases the variance of the angle estimations in an optimal way, improving the interface performance.

This fusion is performed using a decentralized Kalman filter according to ref. [36]. With this aim, the covariance σ^2 computation for each angle β is given by the following recursive equation:[37]

$$\sigma^2(k) = \sigma^2(k-1) + \lambda(\beta(k)^2 - \sigma^2(k-1)),$$

where $\lambda = 0.01$ is the damping factor.

Fusion process is performed because the detection of β_p angle depends on the centroids of the eyes are always detected. This is not always possible due to abrupt changes in the illumination or wide movement of the head. These problems can produce failures in the calculus of the homography. The advantage of this technique is that the angle estimated is the absolute value of the orientation, which gives the global orientation of the head respect axis Y in the 3D space. On the other hand, the optical flow technique used has a lower computing cost. The calculus of the parameters with optical flow does not fail if the illumination changes or if there is wide head movements because the calculation of optical flow does not depend of the geometric points in the space. However, the calculus of optical flow depends on the speed of the turning of head. The problem in this technique is that the angle value β_f obtained is relative, thus producing an accumulation error along the time. This problem is solved setting β_f to zero when both eyes are detected and β_p is zero.

It is important to note that if both techniques are working well, they will provide similar values for the β angle and the output of the Kalman filter will be an intermediate value. On the other hand, if one technique fails, it will provide a sequence of β angles with a large variance, and the output of

Fig. 10. Image sequence where the image processing techniques are applied.

the Kalman filter will be closer to the β angle with minimum variance.

The optical flow and features points on a typical sequence of images are shown in Fig. 10.

2.4. Preliminary experiments

The performance of the developed VBI was first evaluated by 10 expert users in several indoor experiments with different light conditions, i.e., at different times of day. In spite of the constant artificial illumination of the laboratory, light conditions were not always the same for the experiments since the contribution of the natural light were variable. In view of the fact that the developed interface only needs to detect the sign of the α angle but it has to obtain a good estimation of the β angle value, the performance evaluation of the interface was focused on the estimation of the β angle value.

In the carried out experiments, the user starts with a head's orientation equals to zero and then the user turns his head until achieving $\pm 5^{\circ}$, $\pm 10^{\circ}$, $\pm 15^{\circ}$, or $\pm 20^{\circ}$. The ground truth of the head orientation is measured using a laser pointer fixed in the head following the recommendations in ref. [16].

Figures 11 and 12 show some of the most representative results obtained in the experiments carried out. It can be seen in these figures that the accuracy of the proposed sensing method is comparable with the bests similar techniques reported in the literature,[16] with maximum errors below 3°. However, both eyes could be detected even when the deviation of the head is about 30°, but the center of the hidden eye is detected closer to the nose. Therefore, the detecting error is increased.

3. Model of the Robotic Wheelchair

The robotic wheelchair used in this work is the one developed in ref. [38]. The block diagram of this wheelchair is shown in Fig. 13. In this work, the control architecture of the wheelchair includes the kinematics and the dynamic.

The model of the wheelchair is presented in Fig. 14. This figure depicts the wheelchair with the parameters and

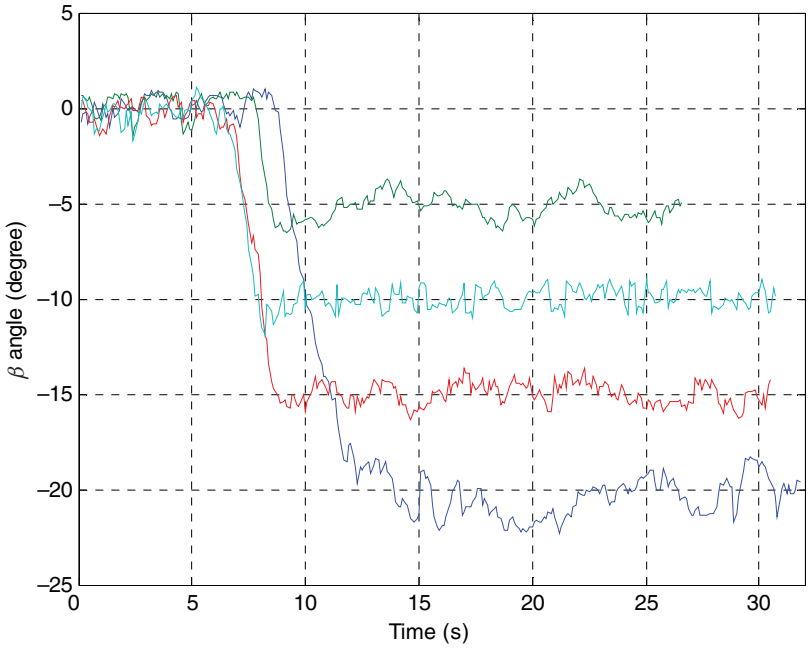

Fig. 11. (Colour online) Estimation β-angle negative values.

variables of interest. In the figure, u and ω are the linear and angular velocities of the wheelchair, respectively. G is the center of mass of the wheelchair, c is the position of the middle point between the front wheels, E is the mass center of the user location, h is the point of interest with coordinate x, y in the XY plane, ψ is the robot orientation, and a is the distance between the point of interest and the central point of the virtual axis linking the traction wheels.

The mathematical representation of the complete model can be seen in the same way of mobile robots[39] and is given by the following:

Kinematic Model:

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} \cos\psi & -a\sin\psi \\ \sin\psi & a\cos\psi \\ 0 & 1 \end{bmatrix} \begin{bmatrix} u \\ \omega \end{bmatrix} + \begin{bmatrix} \delta_x \\ \delta_y \\ 0 \end{bmatrix}. \tag{28}$$

Dynamic Model:

$$\begin{bmatrix} \dot{u} \\ \dot{\omega} \end{bmatrix} = \begin{bmatrix} \frac{\theta_3}{\theta_1}\omega^2 & -\frac{\theta_4}{\theta_1}u \\ -\frac{\theta_5}{\theta_2}u\omega & -\frac{\theta_6}{\theta_2}\omega \end{bmatrix} + \begin{bmatrix} \frac{1}{\theta_1} & 0 \\ 0 & \frac{1}{\theta_2} \end{bmatrix} \begin{bmatrix} u^d_{\text{ref}} \\ \omega^d_{\text{ref}} \end{bmatrix} + \begin{bmatrix} \delta_u \\ \delta_\omega \end{bmatrix}. \tag{29}$$

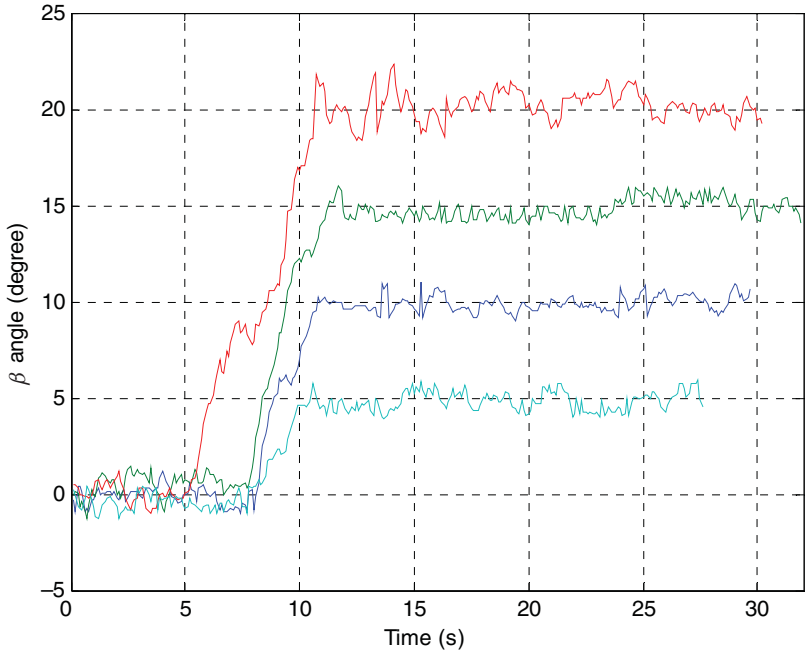

Fig. 12. (Colour online) Estimation β-angle positive values.

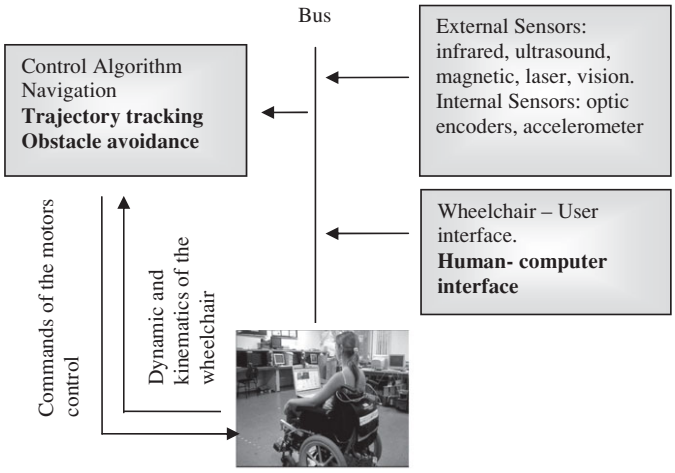

Fig. 13. Block diagram of the robotics wheelchair.

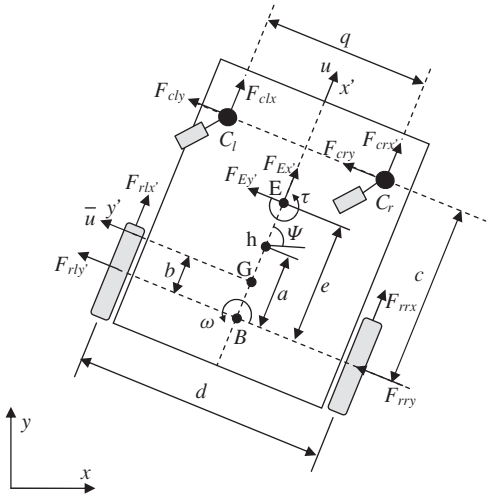

Fig. 14. Model of the robotics wheelchair.

The vector of model parameters and the vector of uncertainties parameters are, respectively,

$$\theta = [\theta_1 \quad \theta_2 \quad \theta_3 \quad \theta_4 \quad \theta_5 \quad \theta_6]^{\mathrm{T}},$$
$$\delta = [\delta_x \quad \delta_y \quad 0 \quad \bar{\delta}_u \quad \bar{\delta}_\omega]^{\mathrm{T}}.$$

Fig. 16. Robotic wheelchair (LAI, Universidade Federal do Espírito Santo, Brazil).

The vector $\boldsymbol{\theta}$ was obtained through an identification experiment as$\theta_1 = 0.4087$; $\theta_2 = 0.1925$; $\theta_3 = 0.0047$; $\theta_4 = 1.0042$; $\theta_5 = 0.0044$; $\theta_6 = 0.8744$.

For more details about the dynamic model, refer to ref. [39].

3.1. Design of the controllers

Based on the kinematic and dynamic model of the robotic wheelchair discussed above, a control system consisting of two controllers is proposed. First one, based on the kinematic model, uses the posture of the user's head obtained by the proposed VBI in order to calculate the linear and angular reference velocities. Second one, based on the dynamic model, calculates the linear and angular velocity commands that ensure that the robotic wheelchair achieves the reference velocities. The inclusion of these controllers allows the user achieving a more safety navigation.[16]

3.1.1. Design of the kinematics controller. The orientation angle β (obtained by the fusion process) of the head commands the angular velocity of the kinematic control, and the angle α (obtained by the optical flow technique) commands the linear velocity of the kinematics control.

The nonlinear control law proposed for the angular velocity is[40]

$$\omega_{\mathrm{ref}} = -k \tanh \tilde{\psi}. \tag{30}$$

This law changes the orientation of the wheelchair. The parameter k is a positive constant, ψ is the orientation of the wheelchair, and β is the reference angle. $\tilde{\psi} = \beta - \psi$ is

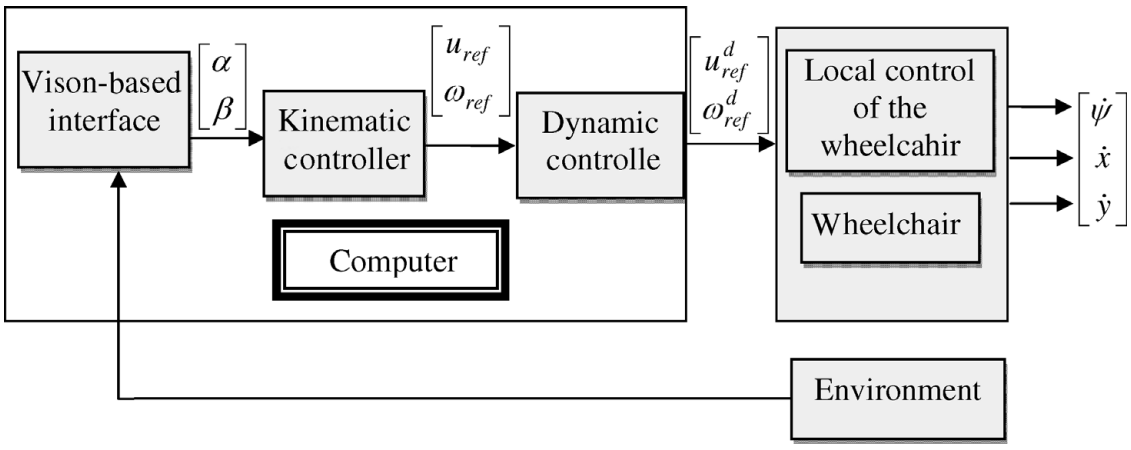

Fig. 15. Blocks Diagram of the architecture of control.

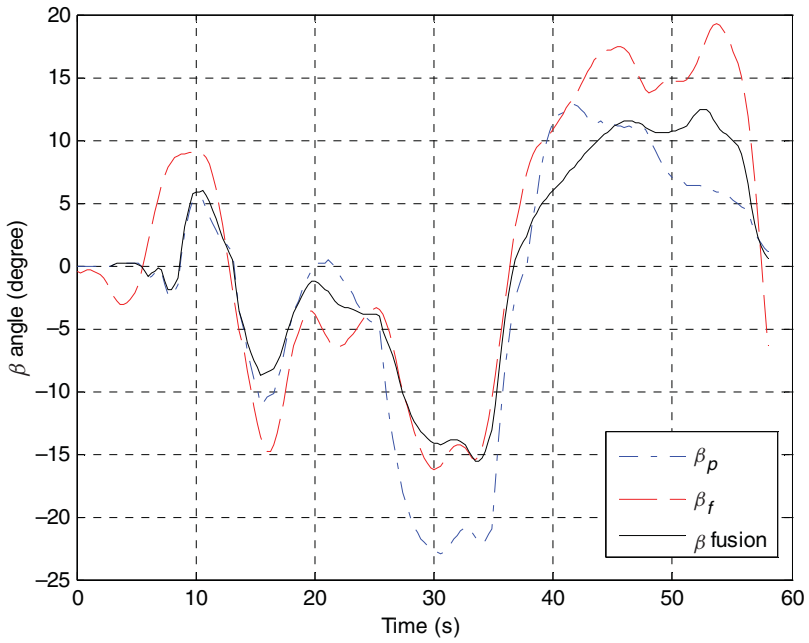

Fig. 17. (Colour online) Variations of the β angle.

the orientation error. The function tanh(.) is used to prevent saturation of commands due to high orientation errors.

The other control command is the linear velocity u_{ref}. This velocity should be decreased when the wheelchair is maneuvering (when $\tilde{\psi}$ is nonzero). The control law proposed is

$$\begin{aligned} u_{\text{ref}} &= V \cos \tilde{\psi} \quad \text{if } \alpha > 0, \\ u_{\text{ref}} &= 0 \qquad\quad \text{if } \alpha < 0. \end{aligned} \tag{31}$$

In this way, the maximum lineal velocity command is $u_{\text{ref}} = V$. Maximum velocity V must be defined taking into account the physic limits of the wheelchair, avoiding actuators saturations. It must be also taken into account in the selection of the value of V the safety and comfort of the user.

3.1.2. Design of the dynamic control. The kinematic control receives the reference signals of the interface (β and α angles). The dynamic controller receives the references of linear and angular velocities, which are generated by the kinematics controller and generates another pair of linear and angular velocities commands to be sent to the wheelchair motors, this shows in Fig. 15.

Then, the dynamic controller is designed based on the robotic wheelchair's dynamics. From Eq. (29) and without considering the uncertainties, the inverse dynamics of the

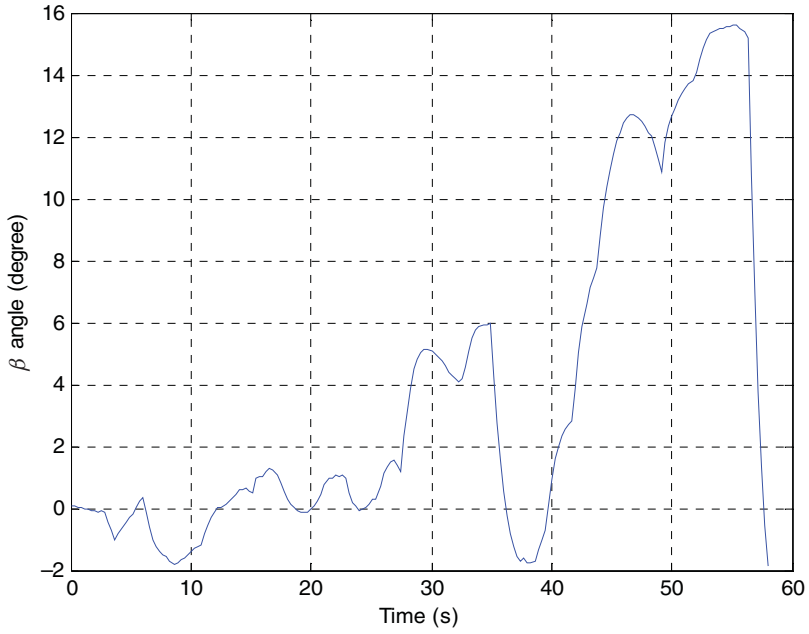

Fig. 18. (Colour online) Variations of the α angle.

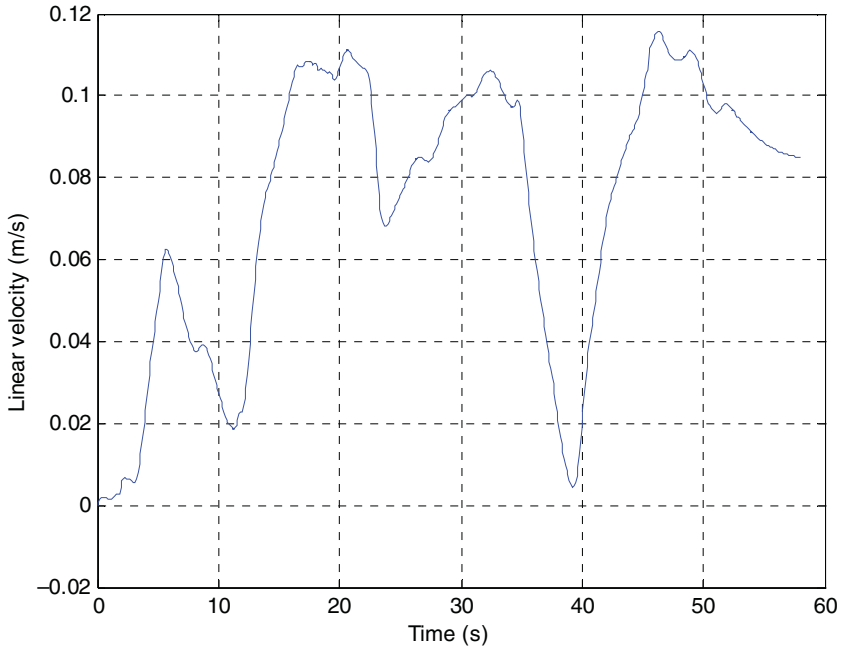

Fig. 19. (Colour online) Linear velocity of the wheelchair.

robotic wheelchair can be parameterized as follows:

$$\begin{bmatrix} u_{\text{ref}}^d \\ \omega_{\text{ref}}^d \end{bmatrix} = \begin{bmatrix} \dot{u} & 0 & -\omega^2 & u & 0 & 0 \\ 0 & \dot{\omega} & 0 & 0 & u\omega & \omega \end{bmatrix} \boldsymbol{\theta}, \tag{32}$$

which can be rewritten as

$$\begin{bmatrix} u_{\text{ref}}^d \\ \omega_{\text{ref}}^d \end{bmatrix} = \begin{bmatrix} \theta_1 & 0 \\ 0 & \theta_2 \end{bmatrix} \begin{bmatrix} \dot{u} \\ \dot{\omega} \end{bmatrix} + \begin{bmatrix} 0 & 0 & -\omega^2 & u & 0 & 0 \\ 0 & 0 & 0 & 0 & u\omega & \omega \end{bmatrix} \boldsymbol{\theta}. \tag{33}$$

The proposed inverse dynamics control law is

$$\nu_{\text{ref}}^d = \mathbf{G}(u, \omega, u_{\text{ref}}, \omega_{\text{ref}}, \dot{u}_{\text{ref}}, \dot{\omega}_{\text{ref}})\boldsymbol{\theta}, \tag{34}$$

where

$$\mathbf{G} = \begin{bmatrix} \sigma_1 & 0 & -\omega^2 & u & 0 & 0 \\ 0 & \sigma_2 & 0 & 0 & u\omega & \omega \end{bmatrix},$$

$$\sigma_1 = \dot{u}_{\text{ref}} + k_u(u_{\text{ref}} - u),$$

$$\sigma_2 = \dot{\omega}_{\text{ref}} + k_\omega(\omega_{\text{ref}} - \omega).$$

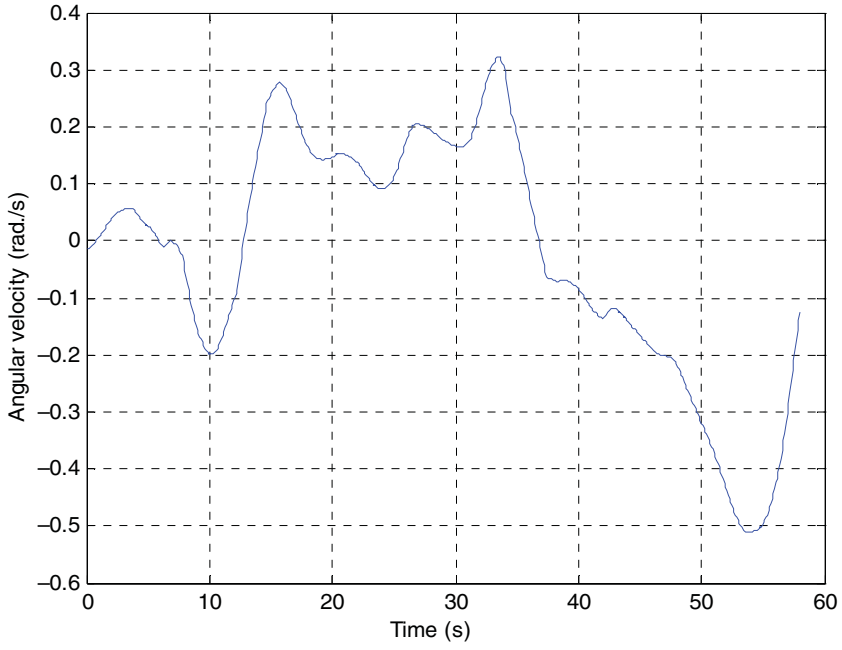

Fig. 20. (Colour online) Angular velocity of the wheelchair.

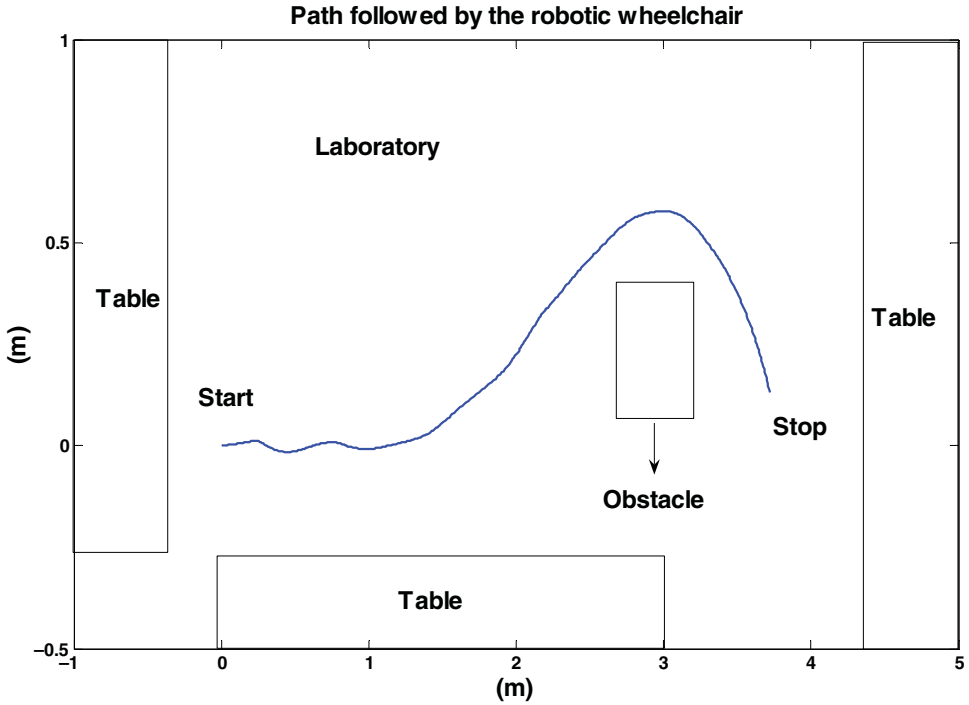

Fig. 21. (Colour online) Path followed by the robotic wheelchair.

4. Experimental Results

Some experiments were carried out using a robotic wheelchair in order to evaluate the performance of the proposed VBI when it is applied to a real assistance device, in real work environments under normal illumination conditions. The VBI was implemented by using a standard webcam (resolution: 320 × 240 pixels; frame rate: 10fps), and it controls the velocity of the wheelchair (see Fig. 16) using a serial port.[38,41] The kinematics control constants are $k = 4$ and $V = 0.1$ m/s, while dynamics control constant are $k_u = 0.9$ and $k_\omega = 0.2$. It is important to remark that the selection of the controller's constant not only depends on the expected performance or physical limit of the actuators, but also (and more important) the comfort and safety of the user. Sometimes, optimal set of constants (from the point of view of the control systems) must be relaxed in order to achieve pleasant navigation, and it depends on each user.

The wheelchair was programmed to move forward when the head rotates forward ($\alpha > 0$, where α being the head rotation angle relative to X-axis), and to stop when the head rotates backward ($\alpha < 0$). The turning of the wheelchair was commanded using β angle, where β being the head rotation angle relative to Y-axis. If the head turned left

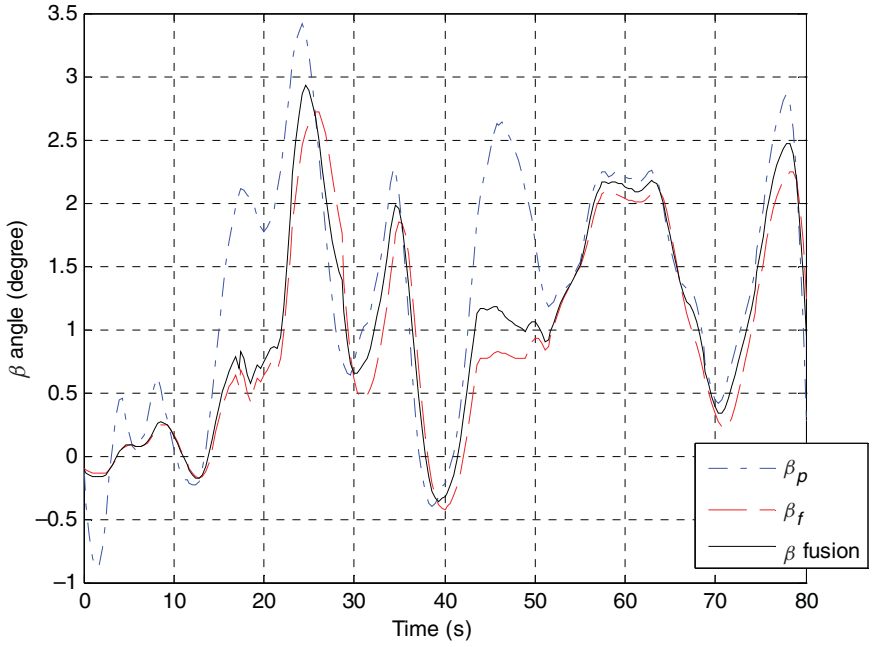

Fig. 22. (Colour online) Variations of the β angle.

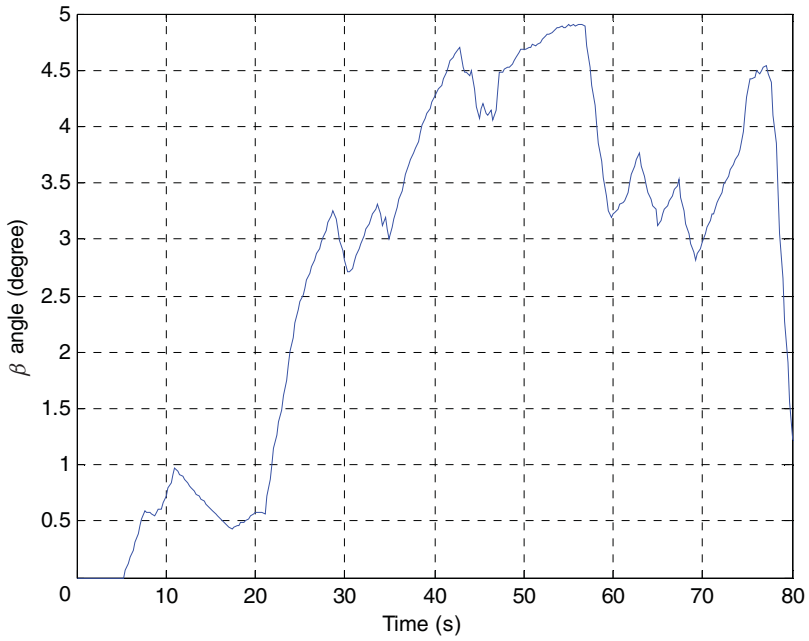

Fig. 23. (Colour online) Variations of the α angle.

($\beta > 0$), the wheelchair turned left; and if the head turned right ($\beta < 0$), the wheelchair was programmed to turn right.

In the first experiment, the user should drive the wheelchair inside a laboratory avoiding obstacles. The results of the experiment can be seen in Figs. 17–21. Figure 17 shows the time evolution of the angle β: dash line refers to the angle obtained by homography; dash-dot line refers to the angle obtained by optical flow; and continuous line denotes the angle obtained by the fusion process. In Fig. 18, the time evolution of α angle can be seen. Linear and angular velocities of the wheelchair are shown in Figs. 19 and 20, respectively. Finally, the trajectory described by the wheelchair is shown in Fig. 21.

In the second experiment, the performance of the interface is evaluated in a corridor, where illumination has larger variations than the other experiments because of the externals source of light. The user drives the wheelchair in a corridor achieving the stable navigation. The results of the experiment can be seen in Figs. 22–26. Figure 22 shows the time evolution of the angle β: dash line refers to the angle obtained by homography; dash-dot line refers to the angle obtained by optical flow; and continuous line denotes the angle obtained by the fusion process. In Fig. 23, the time evolution of α angle can be seen. Linear and angular velocities of the wheelchair are shown in Figs. 24 and 25, respectively. Finally, the trajectory described by the wheelchair is shown in Fig. 26.

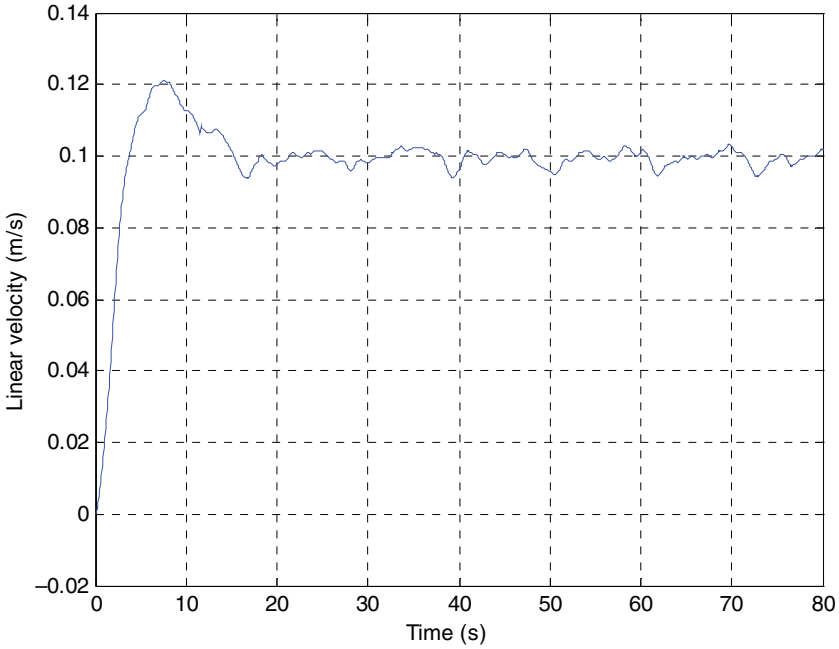

Fig. 24. (Colour online) Linear velocities of the wheelchair.

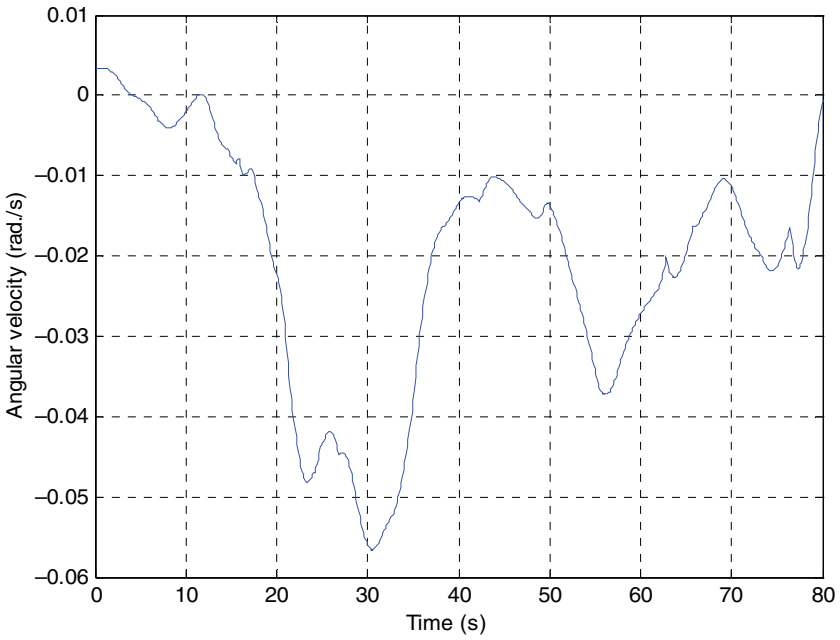

Fig. 25. (Colour online) Angular velocities of the wheelchair.

In the last experiment, the performance of the interface is evaluated in a more complex environment. The user has to drive the wheelchair along four different corridors, so the user has to turn right of left when necessary. Additionally, an obstacle was located in one of the corridors and illumination of the set was altered in order to have different light conditions in the corridors (turning on or turning off some of the lamps). The results of the experiment can be seen in Figs. 27–30. Figure 27 shows the time evolution of the angle β: dash line refers to the angle obtained by homography; dash-dot line refers to the angle obtained by optical flow; and continuous line denotes the angle obtained by the fusion process. Linear and angular velocities of the wheelchair are shown in Figs. 28 and 29, respectively. Finally, the trajectory described by the wheelchair is shown in Fig. 30.

The presented experiments show the good performance obtained with the proposed VBI. It is worth noting that the system's performance has not being altered in spite of the variable light conditions, especially in the second and third experiments. It can be seen that both techniques estimate correctly the angle β even with these light variations.

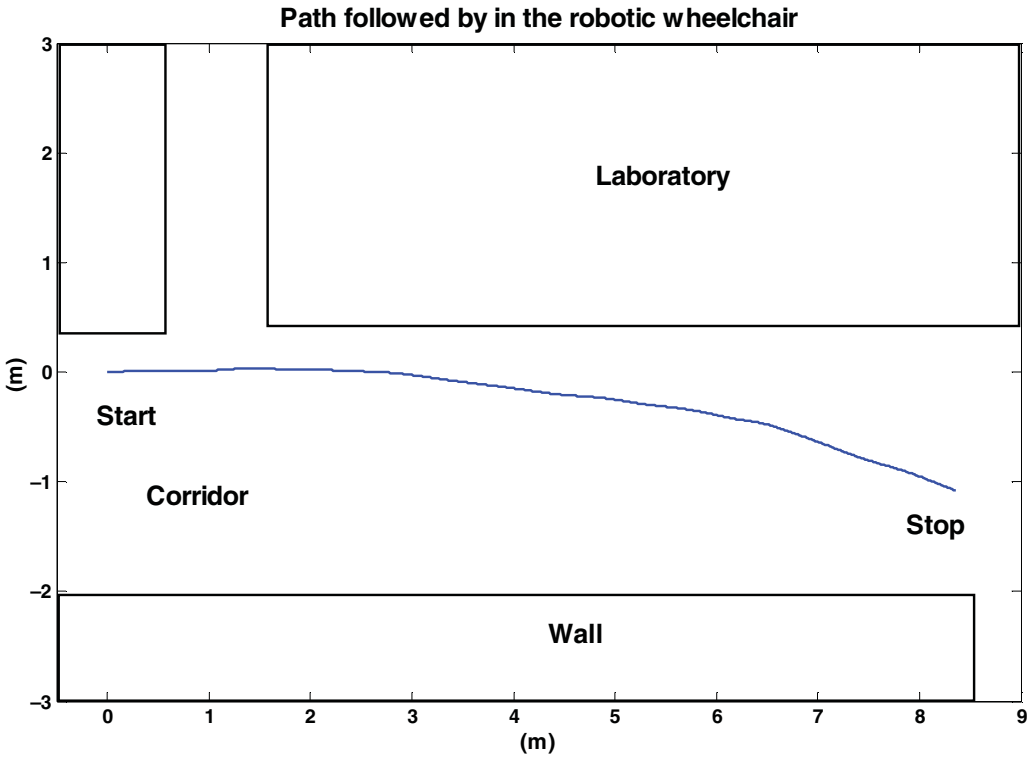

Fig. 26. (Colour online) Path followed by the robotic wheelchair.

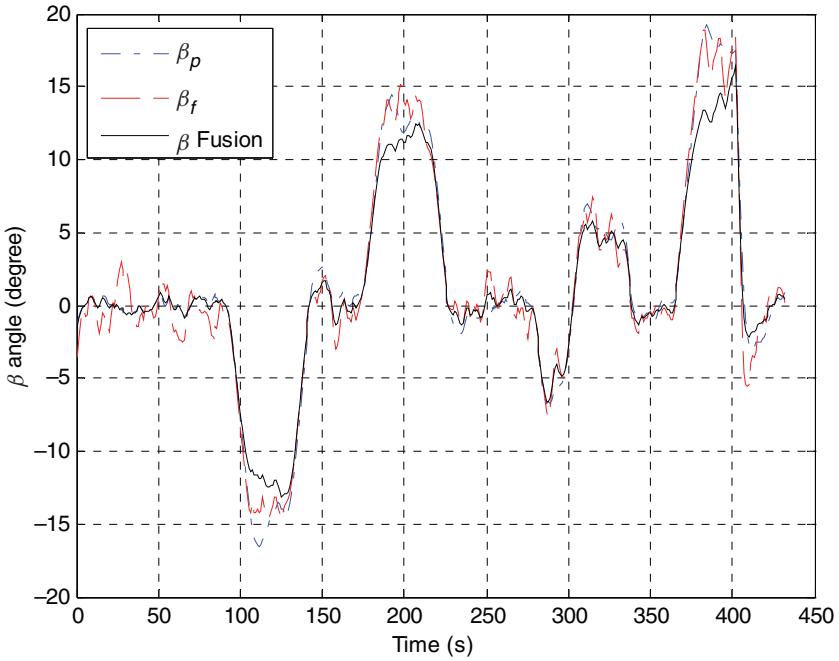

Fig. 27. (Colour online) Variations of the β angle.

It is important to note that experiments were carried out with a trained user, who drives the wheelchair taking care of not deviate his head more than 20° or 25°. However, if an untrained user makes a head movement such that the VBI loses a facial feature, the fusion process becomes important. In this case, the variance of β_p will be large and the output value of the Kalman filter will be closer to β_f. On the other hand, the accumulative error characteristic of optical flow is solved by setting $\beta_f = 0$ when both eyes are detected and $\beta_p = 0$.

The implemented control laws allow achieving a smooth and safety navigation, not only limiting the maximum velocities due to the kinematic controller, but also including a dynamical compensation for reducing the accelerations. Additionally, since linear velocity command depends on the angular error, the control system automatically reduces this linear velocity when a high value of angular velocity is required.

5. Conclusions

A novel assistive robotic system composed by a VBI, a robotic wheelchair and a control algorithm has been presented. This interface estimates the parameters of the user head's pose and translates these parameters into input reference for the controller of a robotic wheelchair.

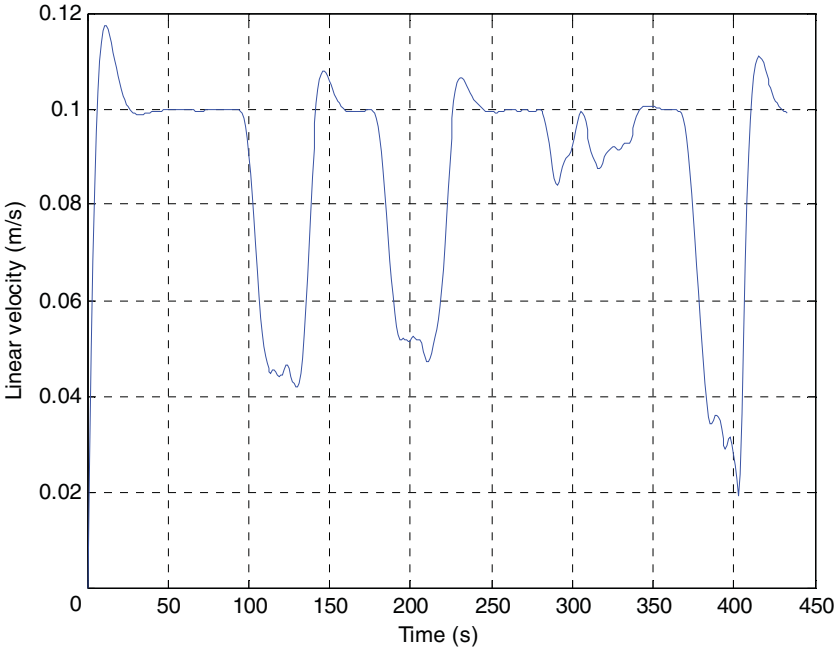

Fig. 28. (Colour online) Linear velocities of the wheelchair.

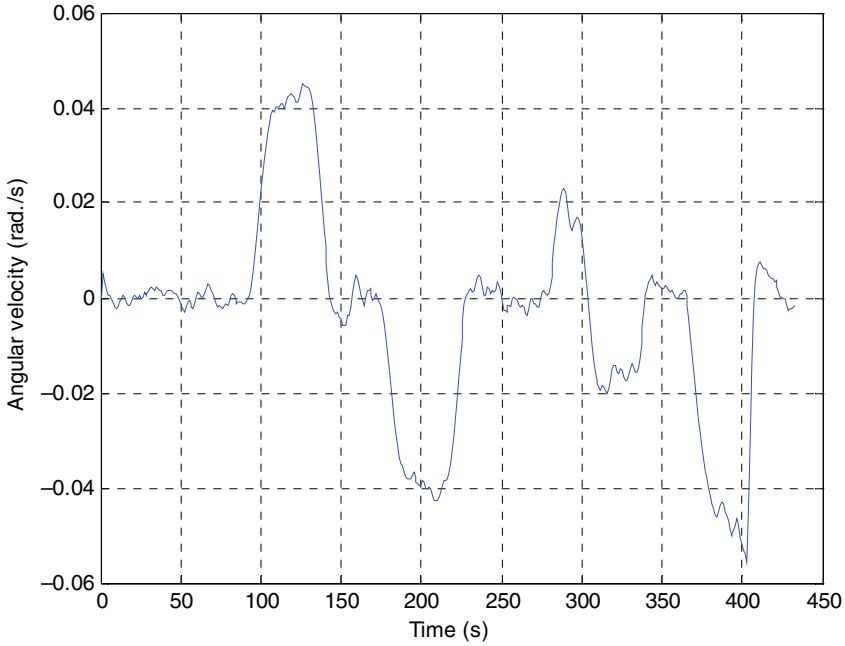

Fig. 29. (Colour online) Angular velocities of the wheelchair.

The performance of the VBI had been evaluated through preliminary static experiments, obtaining maximum errors below 3° for the β angle between $-20°$ and 20°. From these experimental results, it is observed that the fusion of the two techniques improves the performance of the interface and provides better estimations of the β angle. It also provides the redundancy in the system and this is important for the safety of people with high level of disability.

On the other hand, the proposed control system consists in two different cascade subsystems: first, a kinematic-based controller, which generates reference velocities as a function of the head's posture of the user; and the second subsystem, which considers the dynamic model of the wheelchair, calculates the control commands as a function of the velocities reference obtained before.

The overall assistive robotic system had been evaluated. Several indoor experiments with the robotic wheelchair have been carried out, showing the good performance of the proposed system. It is important to stress that experiments were carried out without special the illumination. The ability of the system to perform in these conditions indicates that the system is reliable and robust.

Finally, it is important to remark that in spite of the proposed system is developed for quadriplegic people; it requires that the user can perform voluntary movements of his head. Therefore, as most assistive technology devices, the proposed system cannot be standardized for every person with sever motor disability. The next steps will be focused in the evaluation of the proposed system according to the RESNA-ISO standards, and after that the robotic wheelchair will be tested with motor disable people.

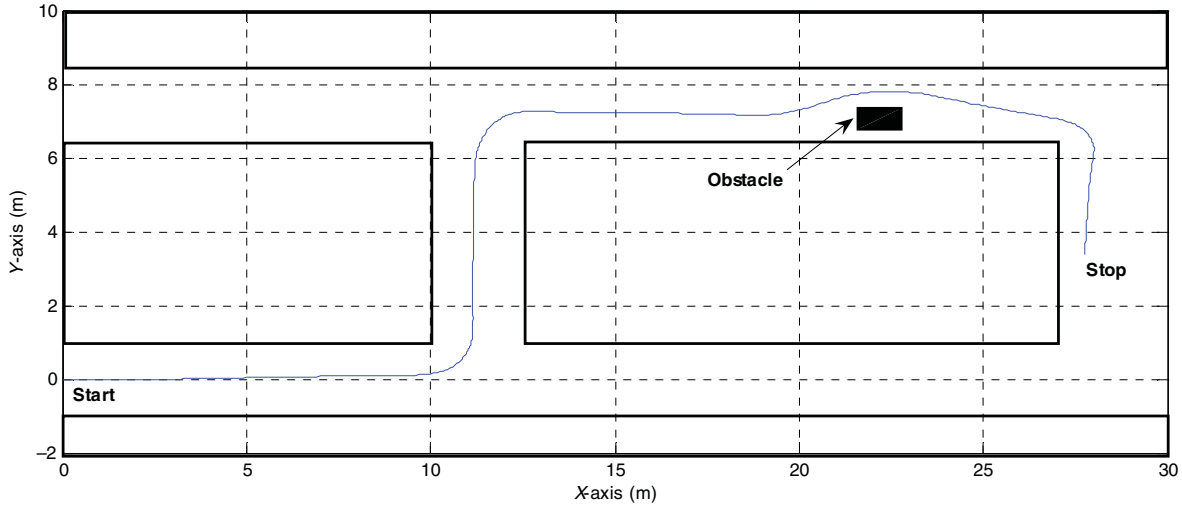

Fig. 30. (Colour online) Path followed by the robotic wheelchair.

References

1. A. Tapus, M. J. Mataric and B. Scassellati, "Socially assistive robotics," *IEEE Robot. Autom. Mag.* **14**, 35–42 (2007).
2. R. M. Mahoney, H. F. Machiel Van der Loos, P. S. Lum and C. Burgar, "Robotic stroke therapy assistant," *Robotica* **21**(1), 33–44 (2003).
3. B. Dellon and Y. Matsuoka, "Prosthetics, exoskeletons and rehabilitation," *IEEE Robot. Autom. Mag.* **14**, 30–34 (2007).
4. H. I. Krebs, B. T. Volpe, M. L. Aisen, W. Hening, S. Adamovich, H. Poizner, K. Subrahmanyan and N. Hogan, "Robotic applications in neuromotor rehabilitation," *Robotica* **21**(1), 3–11 (2003).
5. O. Horn and M. Kreutner, "Smart wheelchair perception using odometry ultrasound sensors and camera," *Robotica* **27**(2), 303–310 (2009).
6. M. Hillman, K. Hagan, S. Hagan, J. Jepson and R. Orpwood, "The Weston wheelchair mounted assistive robot—The design story," *Robotica* **20**(2), 125–132 (2002).
7. M. Ren and H. A. Karimi, "A hidden Markov model-based map-matching algorithm for wheelchair navigation," *J. Navig.* **62**, 383–395 (2009).
8. C. H. Kuo, H. L. Hunag and M. Y. Lee, "Development of agent-based autonomous robotic wheelchair control systems". *Biomed. Eng. Appl. Basis. Comm.* **15**, 223–234 (2003).
9. L. Montesano, J. Minguez, J. M. Alcubierre and L. Montano, "Towards the Adaptation of a Robotic Wheelchair for Cognitive Disabled Children," *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Beijin, China (2006).
10. P. Jia, H. Hu, T. Lu and K. Yuan, "Head gesture recognition for hands-free control of an intelligent wheelchair," *Ind. Robot: Int. J.* **34**, 60–68 (2007).
11. N. T. Nguyen, H. T. Nguyen and S. Su, "Advanced Robust Tracking Control of a Powered Wheelchair System," *Proceedings of the 29th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBS)*, Lyon, France (2007) pp. 4767–4770.
12. Q. Zeng, C. L. Teo, B. Rebsamen and E. Burdet, "A collaborative wheelchair system," *IEEE Trans. Neural Syst. Rehabil. Eng.* **16**, 161–170 (2008).
13. C. Bauckhage, T. Käster, A. M. Rotenstein and J. K. Tsotsos, "Fast Learning for Customizable Head Pose Recognition in Robotic Wheelchair Control," *Proceedings of the Seventh International Conference on Automatic Face and Gesture Recognition (FGR)* (2006).
14. J. S. Ju, Y. Shin and E. Y. Kim, "Vision based interface system for hands free control of an intelligent wheelchair," *J. Neuroeng. Rehabil.* **6**, 33 (2009) (doi:10.1186/1743-0003-6-33).
15. D. Ding and R. A. Cooper, "Electric powered wheelchairs," *IEEE Control Syst. Mag.* **25**(2), 22–34 (2005).
16. E. Muphy-Chutorian and M. Manubhai Trivedi, "Head pose estimation in computer vision: A survey," *IEEE Trans. Pattern Anal. Mach. Intell.* **31**(4), 607–626 (2009).
17. A. Ferreira, W. C. Celeste, F. Auat Cheein, T. F. Bastos Filho, M. Sarcinelli Filho, and R. Carelli, "Human-machine interfaces based on EMG and EEG applied to robotic systems," *J. Neuroeng. Rehabil.* **5**, 1–15 (2008).
18. S. Gong, S. J. MacKenna and A. Psarrou, *Dynamic Vision from Images to Face Recognition* (Imperial Collage Press, London, 2000).
19. F. Dornaika and J. Ahlberg, "Face and facial feature tracking using deformable models," *Int. J. Image Graphics*. **4**, 499–532 (2004).
20. J. B. Chen and E. Tiddeman, "Robust Facial Feature Tracking System, Advanced Video and Signal Based Surveillance," *Proceedings of the IEEE Conference on Advanced Video and Signal Based Surveillance (AVSS '05)*, Italy (2005) pp. 445–449.
21. R. L. Hsu, M. Abdel-Mottaleb, and A. K. Jain, "Face detection in color images," *Trans. Pattern Anal. Mach. Intell*. **24**, 696–705 (2002).
22. K. M. Cho, J. H. Jang and K. S. Hong, "Adaptive skin-color filter," *Pattern Recognit.* **34**, 1067–1073 (2001).
23. M. J. Jones and J. M. Rehg, "Statistical color models with application to skin detection," *Int. J. Comput. Vis.* **46**, 86–96 (2002).
24. V. Bhaskaran and K. Konstantinides, *Image and Video Compression Standards Algorithms and Architectures* (Kluwer Academic Publishers, USA, 1999).
25. S. Z. Li and A. K. Jain, *Handbook of Face Recognition* (Springer Science+Business Media, USA, 2005).
26. M. A. Berbar, H. M. Kelash and A. A. Kandeel, "Face and Facial Features Detection in Color Images," *Proceedings of the Fourth International Conference on Informatics and Systems (INFOS '06)* Cairo, Egipt (2006).
27. B. K. P. Horn, *Robot Vision* (MIT Press, McGraw-Hill, 1986).
28. M. C. Su, K. C. Wang and G. D. Chen, "An eye tracking system and its application in aids for people with severe disabilities," *Biomed. Eng. Appl. Basis Commum.* **8**(6), 319–327 (2006).
29. J. Hannuksela, J. Heikkilä and M. Pietikäinen, "Human-Computer Interaction Using Head Movements," *Proceedings of the Infotech Oulu International Workshop on Processing Sensory Information for Proactive Systems (PSIPS)*, Oulu, Finland (2004).
30. R. Lanzarotti, P. Campadelli and N. A. Borghese, "Automatic Features Detection for Overlapping Face Images on their 3D Range Models," *Proceedings of the 11th International Conference Image Analysis and Processing*, Palermo, Italy (2001) pp. 316–321.
31. S. Z. Selim and M. A. Ismail, "K-means-type algorithms: A generalized convergence theorem and characterization of local optimality," *IEEE Trans. Pattern Anal. Mach. Intell*. **6**, 81–86 (1984).
32. E. Trucco and A. Verri, *Introductory Techniques for 3-D Computer Vision* (Prentice-Hall, New Jersey, 1998).
33. R. Brown and P. Hwang, *Introduction to Random Signals and Applied Kalman Filtering*, 3rd ed. (John Wiley & Sons, New York, USA, 1997).
34. C. M. Soria, R. Carelli and M. Sarcinelli-Filho, "Optical Flow Estimation Using Data Fusion," *VI Simposio Brasileiro de Automacao Inteligente*, Bauru (2003) pp. 259–264.
35. Y. Zhu and K. Fujimora, "3D Head Pose Estimation with Optical Flow and Depth Constraints," *Proceedings of the Fourth International Conference on 3D Digital Imaging an Modeling*, Computer Society IEEE, Banff, Canada (Oct. 6–10, 2003).
36. C. Soria, E. Freire and R. Carelli, "Stable AGV corridor navigation based on data and control signal fusión," *Latin Am. Appl. Res*. **36**(2), 71–78 (2006).
37. P. C. Young, *Recursive Estimation and Time Series Analysis: An Introduction* (Springer Verlag, Berlin, 1984).
38. A. Ferreira; D. C. Cavalieri, R. L. Silva, T. F. Bastos Filho and M. A. Sarcinelli Filho, "Versatile Robotic Wheelchair Commanded by Brain Signals or Eye Blinks," *Proceedings of the International Joint Conference on Biomedical Engineering Systems and Technologies*, INSTICC, Portugal, (2008) vol. 2, pp. 62–67.
39. C. De la Cruz and R. Carelli, "Dynamic model based formation control and obstacle avoidance of multi-robot systems," *Robotica* **26**(3), 345–356 (2008).
40. C. Soria, R. Carelli and M. Sarcinelli-Filho, "Using Panoramic Images and Optical Flow to Avoid Obstacles in Mobile Robot Navigation," *Proceedings of the International Symposium on Industrial Electronics (ISIE)*, Canada (2006).
41. T. F. Bastos Filho, M. Sarcinelli Filho, A. Ferreira, W. C. Celeste, R. L. Silva, V. R. Martins, D. C. Cavalieri, P. N. Filgueira and I. B. Arantes, "Case Study: Cognitive Control of a Robotic Wheelchair," **In**: *Wearable Robots: Biomechatronic Exoskeletons* (J. L. Pons, ed.) Chapter 9, Section 9.6 (Wiley, 2008) pp. 315–319.