Corridor following wheelchair by visual servoing

François Pasteau[1], Marie Babel[1], Rafiq Sekkal[1]

Abstract— In this paper, we present an autonomous navigation framework of a wheelchair by means of a single camera and visual servoing. We focus on a corridor following task where no prior knowledge of the environment is required. Our approach embeds an image-based controller, thus avoiding to estimate the pose of the wheelchair. The servoing process matches the non holonomous constraints of the wheelchair and relies on two visual features, namely the vanishing point location and the orientation of the median line formed by the straight lines related to the bottom of the walls. This overcomes the process initialization issue typically raised in the literature. The control scheme has been implemented onto a robotized wheelchair and results show that it can follow a corridor with an accuracy of ± 3**cm.**

I. Introduction

Navigating within an unknown indoor environment using electrical wheelchair remains a challenging task. Indeed disabled people have to be vigilant while steering the wheelchair, thus inducing fatigability that affects their autonomy. Besides, depending on the disabilities, navigation accuracy can be deeply altered. In particular keeping a stable position along corridors in a secure way and without collisions with walls remains a socially relevant issue.

Wheelchairs observe nonholonomic motion constraints. In this context, researches have been realized for autonomous navigation purposes, especially for path or corridor following tasks. Some related works are based on sonar sensing [1], lidar and stereo cameras [2] or omnidirectionnal cameras [3][4].

In this study, we aim at following a corridor using a single camera fixed onto the rigid body of a wheelchair. In this context, visual servoing frameworks aim at controlling the relevant degrees of freedom (dof) associated with the robot. As for wheelchairs, they belong to differential drive cart like robots, thus requiring two dofs. In addition, they can be modelled as a simple unicycle [5].

In the literature, visual servoing approaches are classically classified into two categories, namely Pose-Based Visual Servoing (PBVS) and Image-Base Visual Servoing (IBVS) [6]. PBVS relies in particular on the geometry of the environment and metrical data, whereas IBVS avoids any explicit representation of the environment. In [7] and [8], it has been shown that image-based controller is more precise and robust when considering path following as well as corridor following. As our study assumes no prior knowledge of the environment, image-based framework is here considered.

Within this scope, visual features have to be carefully chosen to insure autonomous navigation objectives. For indoor environment, especially when dealing with corridor navigation, straight lines associated to vanishing point are relevant visual features [3]. Several studies have been realized in this way. If pose based methods are classically used [1], thus requiring a priori data, few works are devoted to image based solutions in a complete unknown environment. Among the most efficient schemes, in [8], the chosen visual features however induce a lack of robustness as soon as the camera configuration changes, in particular when the orientation of the camera is not aligned with the symmetry axis of the wheelchair.

The visual servoing method proposed in this paper relies on the knowledge of both the vanishing point and the position of the vanishing line that corresponds to the median line in the corridor. To this aim, a dedicated line detection framework has been designed. This image processing process avoids the initialization stage that remains a bottleneck in most of servoing schemes.

The paper is organized as follows. Section II deals with the modelling related to the robotized system. Section III defines the visual features as well as their detection process, while Section IV exposes the control aspects. Visual servoing experiments are described in Section V and results are presented and discussed.

II. Modelling

The wheelchair is considered as a six-wheel robot moving on a horizontal plane. The robot is composed of two differentially actuated wheels located at the middle of the body of the robot, two passive caster front wheels and two passive caster rear wheels.

A wheelchair behaves like a unicycle robot, thus matching nonholonomous constraints. In our context, as the system is aimed at following a corridor (planar ground), we restrict the control variable to the steering (or angular) velocity ω (positive counterclockwise) of the robot, while maintaining constant the forward velocity v. Indeed, when applying a constant forward velocity, the angular velocity variation is sufficient to control the lateral distance of the robot from the walls, as well as its orientation.

We first define the world frame $\mathcal{F}_g(O, x_g, y_g, z_g)$ as presented in Figure 1. A frame $\mathcal{F}_r(P_O, x_r, y_r, z_r)$ is attached to the wheelchair: its origin corresponds to the middle of the segment formed by the centres of the two differentially wheels. $\mathcal{F}_c(C, x_c, y_c, z_c)$ corresponds to the camera frame that is fixed onto the wheelchair, where C denotes the optical center. $\mathcal{F}_I(I_0, X, Y)$ corresponds to the image frame, with I_0 the principal point in the image.

[1]Authors are with INSA Rennes, IRISA and Inria Rennes, France, `marie.babel at irisa.fr`

In $\mathcal{F}_g$, the robot state coordinates are given by

$$q = [x \ y \ \phi]$$

where x and y represent the Cartesian position of the robot, and $\phi \in (-\pi, \pi]$ corresponds to the angle between the driving direction of the wheelchair and the x-axis of the world coordinate frame, that is the positive counterclockwise orientation of the robot with respect to x_g axis (heading angle).

The optical center of the camera is positioned on the wheelchair in such a way that translation $^c\mathbf{t}_r = [w, \ 0, \ -l]^T$. The distance between the floor and the camera optical center is equal to h.

The rotation matrix $^c\mathbf{R}_r$ relative to the fixed orientation of the camera frame with respect to the robot frame is given by

$$^c\mathbf{R}_r = \begin{bmatrix} 0 & -1 & 0 \\ 0 & 0 & -1 \\ 1 & 0 & 0 \end{bmatrix}. \quad (1)$$

The robot jacobian $^r\mathbf{J}_r$ expressed in the robot frame is equal to

$$^r\mathbf{J}_r = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}^T. \quad (2)$$

When considering the control inputs $\mathbf{u} = (v, \omega)$, the kinematic model of the camera expressed in the world frame is given by

$$\begin{cases} \dot{x} = v \cos \phi - l\dot{\phi}, \\ \dot{y} = v \sin \phi - w\dot{\phi}, \\ \dot{\phi} = \omega. \end{cases}$$

III. VISUAL FEATURE EXTRACTION

To realize an image-based servoing of the angular velocity of the wheelchair, two visual features are here proposed.

A. Visual feature definition

First, the x-axis coordinate x_f of the vanishing point f is exploited (see Figure 2). The corridor following task implies that x_f has to be maintained equal to zero, i.e. located at the I_0 point in the image. By doing so, the robot orientation is ensured to be parallel to the walls. As we work in normalised coordinate, the camera focal is considered equal to 1. Then the relationship between the angle ϕ presented in figure 2 and x_f is expressed by $\phi = \arctan(x_f)$.

In addition, the lateral position of the wheelchair within the corridor can also be controlled. The orientations θ_r and θ_l of the straight lines related to the bottom of the walls are here required to compute this position. To this aim, we use the 2D projection of the median line of the corridor into the image. It corresponds to the straight line parametrized by (ρ_m, θ_m) where

$$\rho_m = x_f \cos \theta_m + y_f \sin \theta_m \quad (3)$$

and

$$\theta_m = \arctan \left[\frac{1}{2} \left(\tan(\theta_l) + \tan(\theta_r) \right) \right], \quad (4)$$

as illustrated in Figure 2. In addition, we can observe that the distance y between the robot center and the middle line
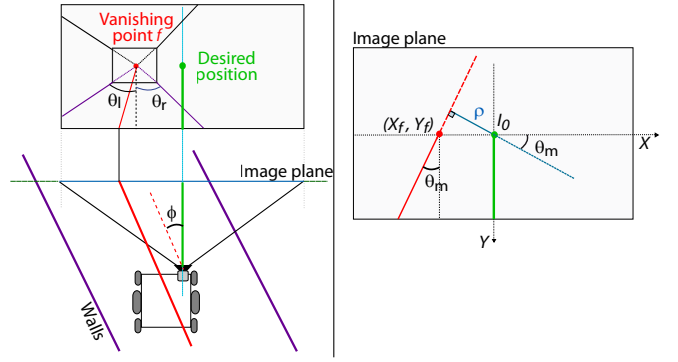

Fig. 2. θ and x_f visual features

of the corridor is such as $y = h * tan(\theta_m)$, where h is the distance between the camera optical center and the floor. In this way, θ_m is chosen as the second visual feature. We can here observe that the corridor width does not need to be estimated to extract these visual feature.

B. Vanishing point estimation

Estimation of vanishing point remains an open issue when aiming at designing an accurate and real-time solution. Vanishing points correspond to locations where a significant number of intersecting straight lines are observed. Classically, vanishing point estimation processes are based on Gaussian sphere projection framework [9], [10]. The idea is to project detected lines in an image onto this sphere. Then the vanishing point is the point which has accumulated the most votes. In addition, to ensure a temporal consistency of the vanishing point, we apply a low-pass filter along the sequence.

To extract straight lines within the image, that are required for vanishing point definition, the LSD (Line Segment Detection) algorithm has been used [11]. Based on local gradient orientations in the image, major segments are detected. These features are then classified into two categories: vertical lines that define infinite vanishing points and non vertical lines. Only these latter are used for the vanishing point estimation framework, thus speeding up the process. To add robustness to this scheme, a merging process is applied to group segments that can be then considered as a single feature. To this aim, for each couple of detected segments, if they share the same slope and their extremities are closed enough to each others, they are merged to form a unique line.

C. Wall/floor line boundary detection

In the literature, different techniques have been proposed for wall/floor boundary detection. In [12] corners corresponding to the intersection of a vertical line and the floor plane are used to define wall/floor boundaries. In [13], a dynamical Bayesian network model is applied on each column of the image to estimate the floor boundary.

In our scheme, in order to detect the wall floor boundary, we first look for a set of non vertical lines as wall/floor boundary. They correspond to vanishing lines that cross the
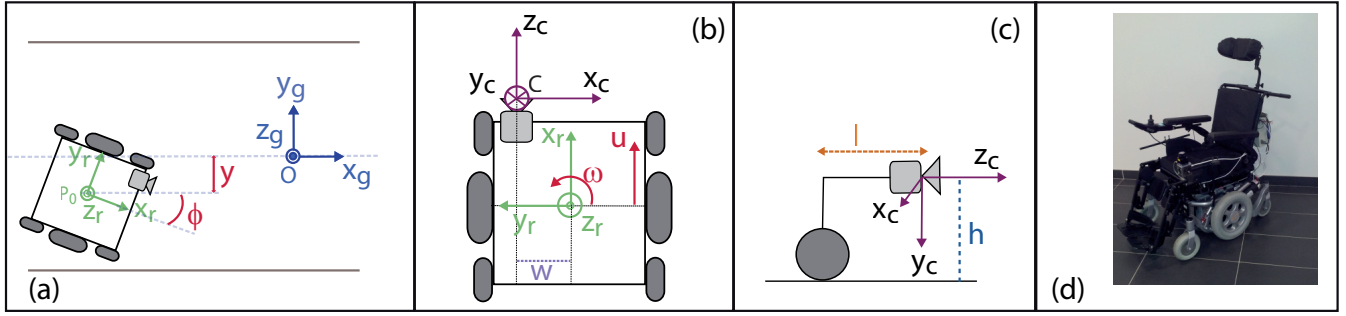
Fig. 1. Frames related to the world, the robot and the camera. (a) Top view of the wheelchair with related global and robot frames. (b) Top view with robot and camera frame relative positions. (c) Simplified side view that characterizes the position of the camera onto the wheelchair. (d) Robotic plateform.

most vertical lines bottom extremities. To minimize false positive detections, a maximal distance between the vertical line extremity and the vanishing line is then defined.

Therefore, from wall/floor line boundary and according to (4), we directly get the value of θ_m.

IV. IMAGE-BASED VISUAL SERVOING

In this section, from both visual features, x_f and θ_m, a control law is designed to achieve an image based visual servoing of the robot. We define $\mathbf{s} = [x_f, \theta_m]$ as the set of visual features to take into account in the servoing task, $\mathbf{s}^* = (0,0)$ as the set of desired visual features and $\mathbf{e} = \mathbf{s} - \mathbf{s}^*$ as the error on visual features. The interaction matrix $\mathbf{L_s}$, which relates the dynamics of visual features to the camera velocity $\mathbf{u}_c$, is given by [14] et [15], where $\lambda_{\theta_m} = (A\sin(\theta_m) - B\cos(\theta_m))/D$ and $AX + BY + CZ + D = 0$ is a plane which includes the line.

$$\mathbf{L_s} = \begin{bmatrix} 0 & 0 & 0 & x_f y_f & -1-x_f^2 & y_f \\ \lambda_{\theta_m} c & \lambda_{\theta_m} s & -\lambda_{\theta_m}\rho_m & -\rho_m c & -\rho_m s & -1 \end{bmatrix}$$
$$c = \cos(\theta_m) \qquad s = \sin(\theta_m) \qquad (5)$$

In our case, the line belongs to the floor with equation $Y = h$. Therefore, we get

$$\lambda_{\theta_m} = cos(\theta_m)/h.$$

From $^c\mathbf{R}_r$ and $^c\mathbf{t}_r$ expressed in Section II, the velocity screw transformation matrix is given by

$$^c\mathbf{W}_r = \begin{bmatrix} ^c\mathbf{R}_r & [^c\mathbf{t}_r]_\times \\ 0_{3\times3} & ^c\mathbf{R}_r \end{bmatrix} \qquad (6)$$

with $[]_x$ the skew matrix expression.

The camera velocity in the camera frame $\mathbf{u}_c$ is derived from the robot velocity in the robot frame $\mathbf{u}$ using

$$\mathbf{u}_c = ^c\mathbf{T}_r\,\mathbf{u}$$

with

$$^c\mathbf{T}_r = ^c\mathbf{W}_r\,^r\mathbf{J}_r, \qquad (7)$$

thus leading to

$$^c\mathbf{T}_r = \begin{bmatrix} 0 & -l \\ 0 & 0 \\ 1 & -w \\ 0 & 0 \\ 0 & -1 \\ 0 & 0 \end{bmatrix}$$

From the two control inputs $\mathbf{u} = (v\,\omega)$, we set $v = const = v^*$ as the linear velocity of the robot is supposed to be constant. Therefore only ω is to be servoed.

According to [7], such a system can be controlled using the control law

$$\omega = -\mathbf{J}_\omega^+(\lambda\mathbf{e} + \mathbf{J}_v v^*). \qquad (8)$$

where

$$\mathbf{J}_\omega = \begin{bmatrix} \mathbf{L}_{x_f}\mathbf{T}_\omega \\ \mathbf{L}_{\theta_m}\mathbf{T}_\omega \end{bmatrix} \qquad (9)$$

$$\mathbf{J}_v = \begin{bmatrix} \mathbf{L}_{x_f}\mathbf{T}_v \\ \mathbf{L}_{\theta_m}\mathbf{T}_v \end{bmatrix} \qquad (10)$$

$\mathbf{L}_{x_f}$ and $\mathbf{L}_{\theta_m}$ correspond respectively to the first and second rows of $\mathbf{L_s}$. $\mathbf{T}_v$ and $\mathbf{T}_\omega$ correspond respectively to the first and second columns of $^c\mathbf{T}_r$.

By combining (7) and (5), we immediately obtain

$$\mathbf{J}_\omega = \begin{bmatrix} 1 + x_f^2 \\ -\lambda_{\theta_m} lc + \lambda_{\theta_m} w\rho_m + \rho_m s \end{bmatrix} \qquad (11)$$

$$\mathbf{J}_v = \begin{bmatrix} 0 \\ -\lambda_{\theta_m}\rho_m \end{bmatrix} \qquad (12)$$

A. Singularities

From equation 11, two singularities of the control law can be observed. If $\phi = -\frac{\pi}{2}$ or $\phi = \frac{\pi}{2}$, then the vanishing point lies at either side of the image plane, with $x_f = \pm\infty$. Therefore, one of the components of J_w tends to $\pm\infty$.

In practice, these two singularities are well known and correspond to cases where the wheelchair is perpendicular to the corridor. In such cases, the robot cannot decide by itself which way to go without any path planning framework. In addition, in the case of a wheelchair, the human should make such a decision.

Moreover, in our experiment, the camera mounted on the robot has a $100°$ field of view. Therefore, if $x_f = \pm\infty$, the vanishing point lies outside the image frame and cannot be detected by the visual feature extraction process, thus leading to avoid aforementioned singularities.

V. VISUAL SERVOING EXPERIMENTS

This section reports the experiments that have been conducted by applying the feature extraction process and control law explained in this paper.

A. Experimental setup

The robot used in our experiments is based on an off-the-shelf Penny and Giles wheelchair adapted to robotic use using RoS middleware [16]. The wheelchair is equiped with one $100°$ field of view, forward looking, AdvanSEE MipSEE camera. The camera exhibits an image resolution of 640×480 pixels and a framerate of 10 frames per second. For security reasons and due to the low framerate of the camera, we set $v^* = 0.2m.s^{-1}$. The camera and system were coarsely calibrated with $h = 0.56m$, $l = 0.38m$, $w = 0.32m$. Prior to the feature extraction, images are rectified against the distortions due to the camera lens.

Visual feature extraction as well as control law computation use the ViSP software [17]. Computation is performed on a Core i7 laptop connected to the wheelchair using Wifi network. With such a configuration a latency of 40 ms is to be expected between the emission of the command from the computer and the application of the command on the wheelchair.

To obtain the wheelchair trajectory independently from the camera, a laser rangefinder, located directly above the camera, is used to estimate the ground truth data. The visual servoing is based on the camera alone, the laser rangefinder is then only used for validation purposes as explained in the following section.

To illustrate the validity of the approach used in this paper, various experiments have been carried out. The one presented here has been realized in one of the corridors of Inria building in Rennes, France. The proposed visual servoing is aimed so that from an unknown position, the wheelchair bot will follow the corridor while maintaining the camera in the middle between the two walls and keeping an aligned heading with the corridor.

The experimentations have been realised using the following protocol:

- first, the wheelchair is located at one side of a corridor with an unknown heading and relative position of the camera to the wall,
- then, visual servoing is applied to the wheelchair so that the robot will center itself between the two walls and aligned itself with the corridor,
- the experimentation lasts until the end of the corridor is reached to illustrate the stability of the vanishing point detection,
- the servoing is stopped at the end of the corridor.

No initialisation of the algorithm is required, therefore human intervention is needed only to turn on and off visual servoing at the beginning and at the end of the experiment.

B. Ground truth

In our experiments a Hokuyo URG-04LX-UG01 rangefinder is used. Such a rangefinder exhibits a sampling of $0.36°$ between measurements and a precision of $\pm30mm$ for ranges less than $1000mm$ and $\pm3\%$ otherwise.

To estimate the pose of the wheelchair, the corridor walls have to be first detected. Such a task is achieved by using

Hough transform on the laser output and by discriminating the two longest lines in the image.

Orientation ϕ as well as the position Y of the wheelchair in the corridor can then be obtained from the orientation and the position of the two detected lines.
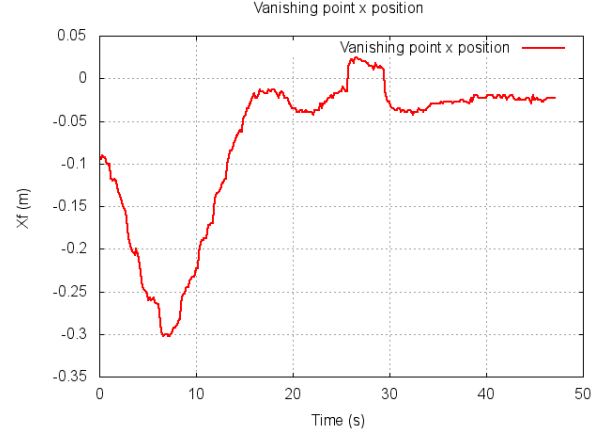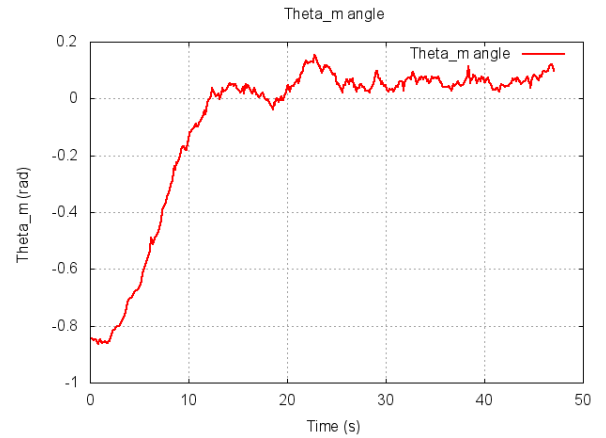

Fig. 3. x_f vanishing point feature

Fig. 4. θ_m line feature

C. Results

Figures 3 and 4 present the evolution of the two visual features, respectively x_f and θ_m during the experimentation.

As explained in the previous section, from the laser rangefinder, ϕ and r are estimated. Figures 6 and 7 presents the evolution of, respectively, ϕ and r.

From the visual features x_f and θ_m, an angular velocity command is computed according to (8). Figure 5 presents the evolution of the computed angular velocity command sent to the wheelchair control unit.

Figure 8 presents the main steps of the visual servoing experiment. The first row of images presents the current images acquired by the camera. The desired feature line is shown in green while the current line is shown in red. Both
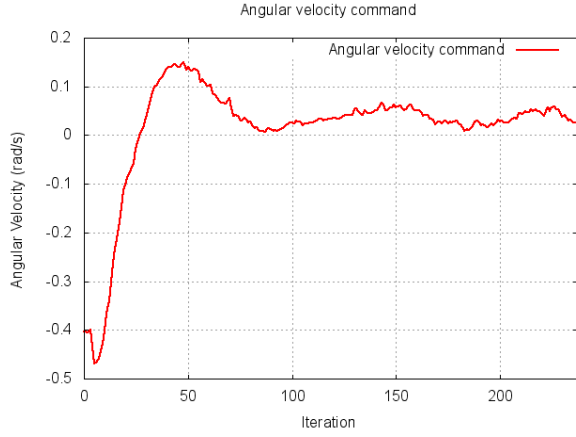
Fig. 5.   Angular velocity command applied to the wheelchair robot

Fig. 7.   Position of the wheelchair relative to the median line of the corridor
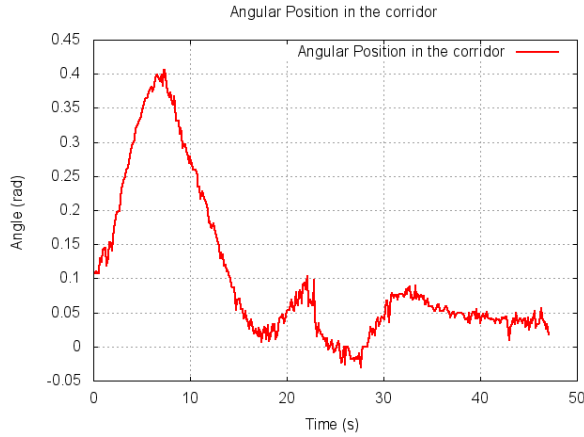

Fig. 6.   Orientation in the corridor

estimated wall/floor boundaries are represented in purple. Finally the desired and current estimated vanishing points are respectively represented by a green and a red crosses. The second row of images corresponds to the reconstructed laser outputs with Hough transform line detection. The third row illustrates the position and orientation of the wheelchair in the corridor at each iteration.

First, the wheelchair is located at one side of the corridor (Figure 8a). From the reconstructed laser output (Figure 8e) and Figure 7, one can observe that the start position of the wheelchair is close to left wall of the corridor. In addition, the robot is slightly oriented toward the wall as shown on Figure 8i and 6. During the experimentation, both of the features tend to 0. However a static error on the x_f vanishing point position can be observed. This static error can be explained by the behavior of the caster wheels. In fact, caster wheels tend to constraint the motion capability of the wheelchair when small changes of orientation are needed, which is usually the case in steady state. The effect of the caster wheels can also be observed on Figure 5. In steady state, the applied command is different from $0 rad/s$

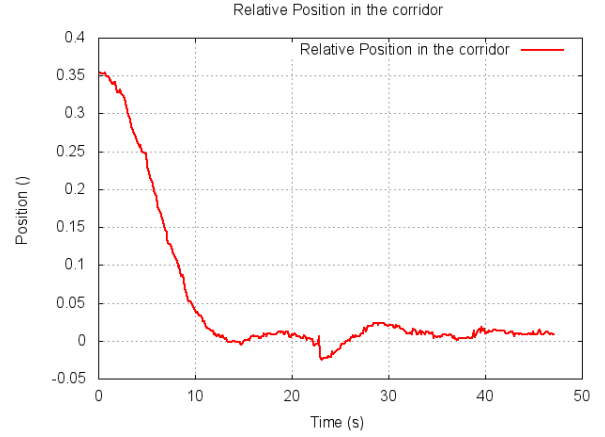as small values are filtered out by the caster wheel behavior. Moreover, the latency between the emission of the command from the computer and its application on the wheelchair as well as the instability of WiFi lead to some pertubations in steady state.

From Figure 7, in steady state, the error on the position wheelchair robot is around $\pm 2\%$ which corresponds to around $3.2 cm$ for $1.6 m$ wide corridor. These results validate the use of our visual feature extraction process for corridor following applications.

VI. CONCLUSION

In this paper, we proposed a complete solution for corridor following wheelchairs. First, two visual features were selected, i.e. the vanishing point x_f position and the angle θ_m obtained from the projection of the median line of the corridor onto the image. An automatic extraction process from camera images of these visual features was realized. A control law was then designed, taking into account the kinematic of the wheelchair.

Experiments have been conducted on the robotized wheelchair. Results exhibit an accurate following of the middle of the corridor with an accuracy of $\pm 2\%$ of the corridor width. Future works aim at taking into account the dynamic model of the wheelchair, including the weight of the passenger as well as the behavior of caster wheels. This solution will be soon tested in real life condition with disabled people thanks to Ergovie company.

REFERENCES

[1] R. Carelli and E. O. Freire, "Corridor navigation and wall-following stable control for sonar-based mobile robots," *Robotics and Autonomous Systems*, vol. 45, no. 34, pp. 235 – 247, 2003.
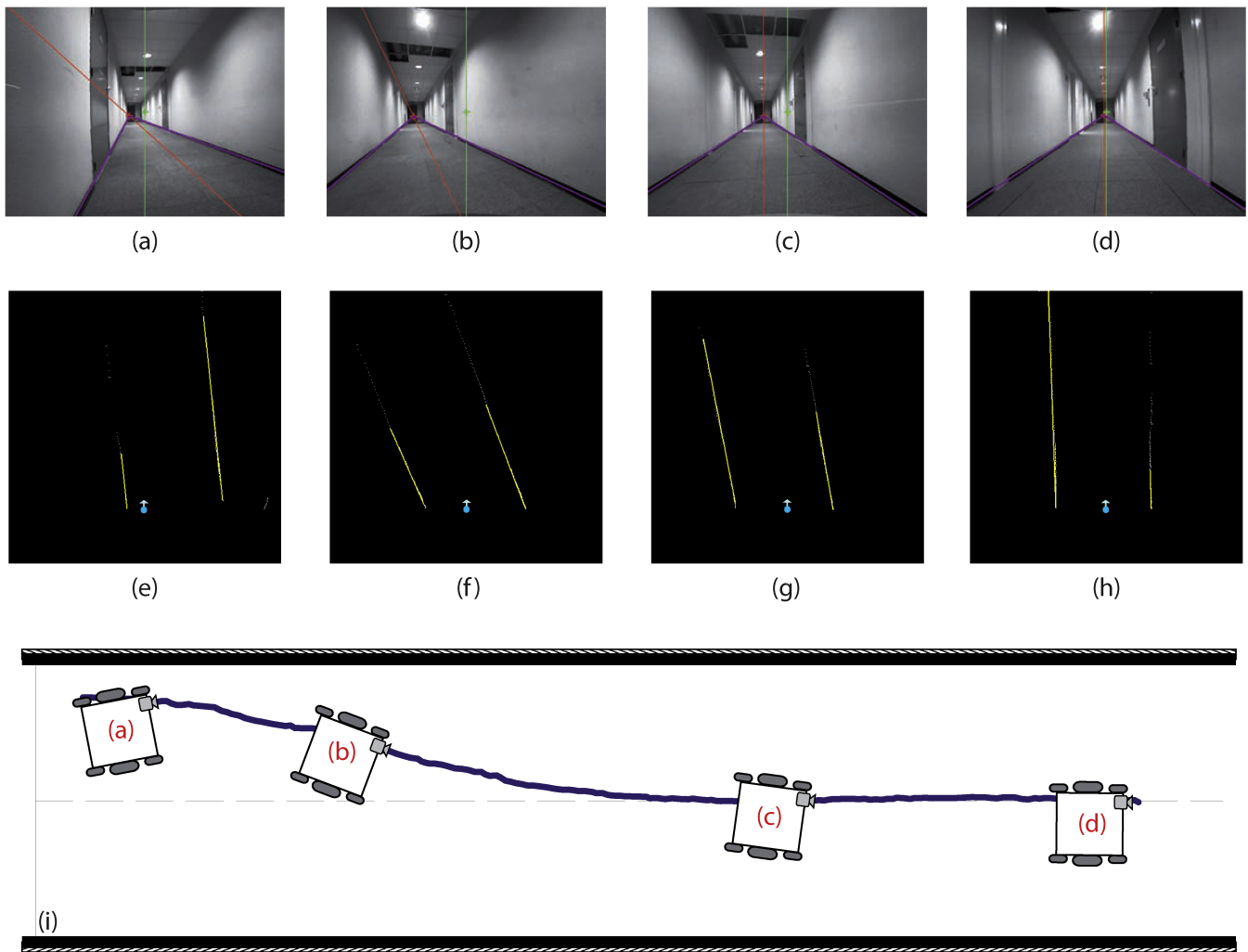
Fig. 8.   Position of the wheelchair relatively to the walls. (a)-(d) Camera images with vanishing point (red) and desired line (green). (e)-(h) Laser outputs (in white) with Hough transform line detection (in yellow). Resulting wheelchair trajectory is depicted below and the position of the robot is modelized.

[2] A. Murarka, S. Gulati, P. Beeson, and B. Kuipers, "Towards a safe, low-cost, intelligent wheelchair," in *Workshop on Planning, Perception and Navigation for Intelligent Vehicles (PPNIV)*, 2009, pp. 42–50.

[3] H. Hadj-Abdelkader, Y. Mezouar, P. Martinet, and F. Chaumette, "Catadioptric visual servoing from 3-d straight lines," *IEEE Transactions on Robotics*, vol. 24, no. 3, pp. 652–665, June.

[4] S. Benhimane, E. Malis, P. Rives, and J. R. Azinheira, "Vision-based control for car platooning using homography decomposition," in *IEEE Inter. Conf. on Robotics and Automation ICRA 2005*.   IEEE, 2005, pp. 2161–2166.

[5] J. J. Park and B. Kuipers, "A smooth control law for graceful motion of differential wheeled mobile robots in 2d environment," in *IEEE Inter. Conf. on Robotics and Automation ICRA 2011*, May, pp. 4896–4902.

[6] F. Chaumette and S. Hutchinson, "Visual servo control, part i: Basic approaches," *IEEE Robotics and Automation Magazine*, vol. 13, no. 4, pp. 82–90, December 2006.

[7] A. Cherubini, F. Chaumette, and G. Oriolo, "Visual servoing for path reaching with nonholonomic robots," *Robotica*, vol. 29, no. 7, pp. 1037–1048, December 2011.

[8] J. Toibero, C. Soria, F. Roberti, R. Carelliz, and P. Fiorini, "Switching visual servoing approach for stable corridor navigation," in *Inter. Conf. on Advanced Robotics, ICAR 2009*, June, pp. 1–6.

[9] C. Rother, "A new approach for vanishing point detection in architectural environments," in *In Proc. 11th British Machine Vision Conference*, 2000, pp. 382–391.

[10] K. Boulanger, K. Bouatouch, and S. Pattanaik, "ATIP: A Tool for 3D Navigation inside a Single Image with Automatic Camera Calibration," in *EG UK conference*, 2006.

[11] R. Grompone von Gioi, J. Jakubowicz, J.-M. Morel, G. Randall, "LSD: a Line Segment Detector," *Image Processing On Line*, 2012.

[12] K. Ok, D.-N. Ta, and F. Dellaert, "Vistas and wall-floor intersection features - enabling autonomous flight in man-made environments," in *Workshop on Visual Control of Mobile Robots*, 2012.

[13] E. Delage, H. Lee, and A. Ng, "A dynamic bayesian network model for autonomous 3d reconstruction from a single indoor image," in *, IEEE Conf. on Computer Vision and Pattern Recognition*, vol. 2, 2006, pp. 2418 – 2428.

[14] J. R. Azinheira and P. Rives, "Image-based visual servoing for vanishing features and ground lines tracking: Application to a uav automatic landing," *International Journal of Optomechatronics*, vol. 2, no. 3, 2008.

[15] B. Espiau, F. Chaumette, and P. Rives, "A new approach to visual servoing in robotics," *IEEE Transactions on Robotics and Automation*, vol. 8, no. 3, pp. 313–326, 1992.

[16] M. Quigley, K. Conley, B. P. Gerkey, J. Faust, T. Foote, J. Leibs, R. Wheeler, and A. Y. Ng, "Ros: an open-source robot operating system," in *ICRA Workshop on Open Source Software*, 2009.

[17] E. Marchand, F. Spindler, and F. Chaumette, "ViSP for visual servoing: a generic software platform with a wide class of robot control skills," *IEEE Robotics and Automation Magazine*, vol. 12, no. 4, pp. 40–52, 2005.