AN EFFECTIVE LOCALIZATION METHOD FOR ROBOT NAVIGATION THROUGH COMBINED ENCODERS POSITIONING AND RETIMING VISUAL CONTROL

Received 7th October 2008; accepted 23rd December 2008.

Mohamed Jallouli, Lobna Amouri, Nabil Derbel

Abstract:

The paper presents an optimal mobile robot localisation method through encoders' measurements and absolute localization using webcam data. This technique has been developed and implemented for the motion of the robot from an initial position towards another desired position, taking into account Kinematic constraints. The proposed method is a cheap technique, which deals with the problem of robot initial position as the visual system provides the origin absolute coordinates related to the reference system. First, we have developed an interface, which ensures the real-time localisation of the mini robot Khepera II while tracking a virtual moving target. Secondly, we have carried out experimentations of both the relative localization method, which determines the speed values of each driving wheel, and absolute localization using webcam data, which determines the robot start position. Next, a correction of the mobile robot position has been realized by retiming points using webcam measures. The obtained results are compared and discussed through different trajectories.

Keywords: *localization, encoders' measurements, fuzzy controller, webcam data, segmentation, retiming method.*

1. Introduction

During the last decades, the useful range for robots has gradually spread to a wide variety of areas. Mobile robots are especially being used as a substitute for humans or to do simple work that is either in or outside. In such a mobile robot system, getting exact information on its current position is very important. At all times, the mobile robot must know instantaneously its current position and the one of the objective. There are two position-estimation methods applied in navigation systems, i.e. absolute and relative positioning.

Relative localization is realised through measurements provided by sensors measuring internal variables of the vehicle. The incremental encoders are the typical inertial sensors. These sensors placed on the wheels axis, which represents the rotation axis of the vehicle. The diesadvantage of this method is that errors of each measurement are accumulated. This heavily degrades the estimates of the position and the orientation of the vehicle, especially for long and winding trajectories [1]. However, absolute localization is based on the use of sensors measuring some parameters of the environment in which the robot is operating. A set of sonars is generally used as an external sensory device. The infrared sensors are implemented on the robot and measure the distance with the environment. These sensors are also widely utilized for the guidance of autonomous vehicles with obstacle avoidance in unknown environment [2, 3]. The major disadvantage of absolute measurements is their dependence on the characteristics of the environment.

In order to compensate these drawbacks, the localization method that fuse data coming from odometers and sonar sensors by applying Hybrid Kalman Filter Fuzzy Logic Adaptive Multisensor Data Fusion Architectures [4, 5] can be used. An other approach adapts the position and the orientation of a mobile robot through a weighted Extended Kalman Filter (EKF) [6, 7]. These methods need much calculation for a mobile robot to perform a task. Other disadvantages are either the short range of used sensors or the necessity to know the initial position of the robot.

Other solution uses a method that estimates a position of a robot using a CCD camera fixed on ceiling of the corridor by calculating the distance moved and time of a mobile robot [8]. Another way that is presented for a robot equipped with a CCD camera calculates its position by recognizing a characteristic topography and compares it with the model image saved in memory [9]. A last proposed method calculates the position of the robot in order to intercept a moving target through visual feedback [10]. The most important disadvantage of these methods is the necessity to know the initial position of the robot.

In this paper, the proposed method uses two different sensors. First, with encoders, which ensure relative positioning using the kinematic model, a fuzzy controller is used to attempt a virtual moving target. Second, a camera installed on the ceiling of the test environment ensures the absolute localization using a segmentation method and a location algorithm and reduces the encoder position errors regardless of a retiming visual control. The experimental results show the effectiveness of the proposed algorithm. The system compensates for robot positioning by means of the following sequences. First, the system calculates the initial position and the orientation by using webcam data. Secondly, the fuzzy controller generates velocities to be applied on the mobile base during an optimal trajectory from the initial position towards another desired position, taking into account kinematic constraints. During the robot navigation, the system calculates its position using encoders' measurements and webcam data. The system compensates encoder errors by retiming point using the visual controller. This article is organized in the following way. In section 2, we present the relative localization system studied with mathematical formulation of the problem and the optimised fuzzy controller design. Section 3 describes the absolute localization system used with the position and the orientation estimation algorithm. Section 4 shows the experimental tests applied to the mobile base *Khepera II*.

2. Relative localization

In this work, relative localization is realised through measurements provided by the encoders fixed on the mobile robot wheels and used both the kinematic model and the fuzzy controller system.

2.1. The mathematical formulation

Several types of mobile robots with driving wheels and encoder system have been studied in the literature [11, 12]. The most studied types are those of the steering angle commanded vehicles [13, 14, 15]. However, our experiments will be carried up onto a mobile robot with two independent driving wheels, which can be oriented and commanded by acting on the speed of each wheel, as shown on the schematic model (Fig. 1).

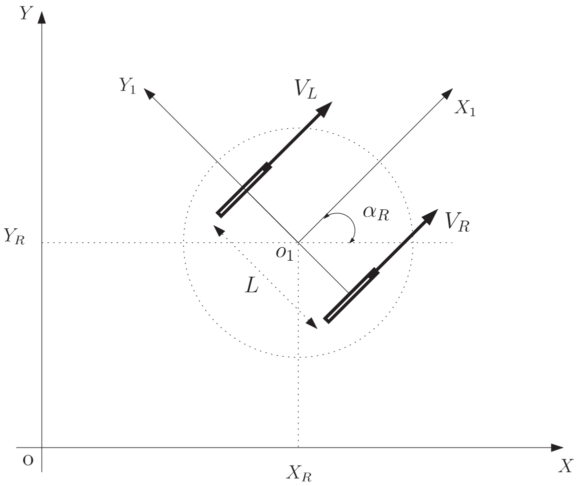


Fig. 1. The schematic model of a nonholonomic mobile robot.

The kinematic model is given by:

$$\begin{cases} \dfrac{X_R}{dt} = \dfrac{V_R + V_L}{2}\cos\alpha_R \\ \dfrac{Y_R}{dt} = \dfrac{V_R + V_L}{2}\sin\alpha_R \\ \dfrac{d\alpha_R}{dt} = \dfrac{V_R - V_L}{L} \end{cases} \quad (1)$$

where V_R and V_L are the robot's right and left wheel's velocities, respectively; $\dot{\alpha}_R$ is the robot's angular velocity, L is the distance between two wheels and α_R is the angle between the robot's direction and the X-axis. By discretization of the system (1) using Euler method, it becomes:

$$\begin{cases} X_R^{new} = X_R^{old} + T\dfrac{V_R^{old} + V_L^{old}}{2}\cos\alpha_R^{old} \\ Y_R^{new} = Y_R^{old} + T\dfrac{V_R^{old} + V_L^{old}}{2}\sin\alpha_R^{old} \\ \alpha_R^{new} = \alpha_R^{old} + T\dfrac{V_R^{old} - V_L^{old}}{L} \end{cases} \quad (2)$$

where T is the sampling time.

To further optimise the mobile trajectory, we propose a fuzzy controller calculating the velocities to be applied on the robot wheels.

2.2. Fuzzy controller synthesis

The most employed strategies for the mobile robot navigation uses Fuzzy Logic Controllers (FLC). The contribution of our study is to optimise a fuzzy controller by means of the gradient method and its validation by the implementation onto the mobile robot Khepera II. The fuzzy controller allows the robot to reach a virtual moving target point starting from a given position. The robot position and orientation are calculated in real time by the odometer module. The controller outputs are velocities V_R and V_L applied respectively on the left and the right driving wheels of the mobile robot, in order to reach a desired position. A bloc diagram of the fuzzy logic controller is shown in Fig. 2.

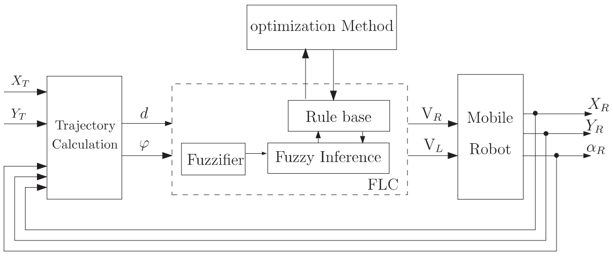


Fig. 2. Block diagram of the fuzzy logic controller optimised by Gradient Method.

The controller has two inputs the distance d and the angle φ. Where d is the distance between the centre of the robot and its target; and φ is the difference between the angle of robot's direction and the angle of the connecting line between the centre of the robot and its target (see Figure 3). The distance d and the angle φ are expressed as follows:

$$d = \sqrt{(X_T - X_R)^2 + (Y_T - Y_R)^2} \quad (3)$$

$$\varphi = \theta_T - \alpha_R \quad (4)$$

with

$$\theta_T = tan^{-1}\frac{(Y_T - Y_R)}{(X_T - X_R)} \quad (5)$$

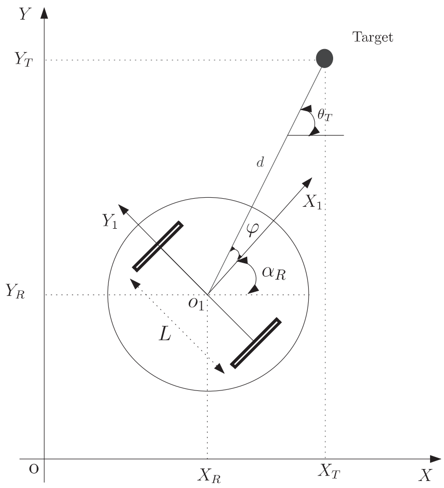


Fig. 3. Robot configuration according to the objective.

From our experiments, we construct five membership functions for the distance (d) and four for the angle (φ). $\mu_{ij}(d)$ and $\mu_{ij}(\varphi)$ are fuzzy subsets defined by their corresponding membership functions, i.e., μ_{Ai} and μ_{Bi}. Five fuzzy subsets are assigned to the variable distance (d): VS: Very Small; S: Small; M: Medium; L: Large; VL: Very Large. Four fuzzy subsets have been associated to the angle (φ): Z: Zero; PS: Positive Small; PM: Positive Medium; PL: positive Large. For each input variable value combination, an action of the output variables is associated to it. Fuzzy rules (situation/action) are proposed in table 1. These rules are manually constructed following several simulations. With the notations: V_L: Left Velocity, V_R: Right Velocity, B: Big, VB: Very Big, S: Small, Z: Zero, M: medium.

Table 1. Linguistic inference table.

V_L / V_R		φ							
		Z		PS		PM		PL	
		VL	VR	VL	VR	VL	VR	VL	VR
d	VS	Z	Z	Z	S	S	M	S	B
	S	S	S	S	M	S	B	M	VB
	M	M	M	S	B	M	VB	VB	VB
	L	B	B	M	VB	B	VB	B	VB
	VL	VB	VB	B	VB	B	VB	B	VB

3. Absolute localization

In this proposed method, the absolute localization is based on the use of a webcam, which measures some parameters of the environment in which the robot is operating. This approach is based on four steps. The first one consists on the image pre-processing. The second one is an orientation estimation with the external sensor. In the third step, we propose an optimising pre-processing method in order to reduce the time processing and to ensure real time localization. In the fourth and last step, we present the transformation from image coordinates to real coordinates algorithm.

3.1. Image pre-processing

We extract the pixel coordinates of the reference system using region properties of a reference image, which is captured and stored in advance. This image describes the workspace of the mobile robot limited by four landmarks placed on the corners. Figures 4 and 5 describe this reference image. In this paper, we apply a binary mask for removing illumination image noises, and selected images, which have 640*480 pixels for an image. The filtering method that has been adopted is a morphological one. Through labelling, we separate objects and search for their features [16]. Then, to recognize whether object is the robot or not, we use region properties (areas and centroid) of an input image which is being input consecutively.

3.2. Orientation Estimation

In this section, we will introduce a technique used in order to calculate the orientation of the mobile robot. We have attached a white landmark on the robot. This landmark is situated in the x-axis of the robot reference system as shown in the robot configuration in Figure 6. The estimation of the robot orientation is based on the pixel coordinates of the landmark centroid. These coordinates are reached using webcam data and the pre-processing method described above. Figure 7 shows a binary image with robot and landmark centroids.

The orientation angle is calculated beyond of the robot pixels coordinates as shown in Figure 8. So, by a simple difference between the robot and the landmark pixel coordinates, we obtain the orientation variation along the x-axis and the y-axis as described in the following equation.

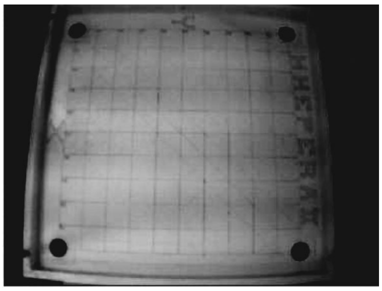


Fig. 4. Reference Image.

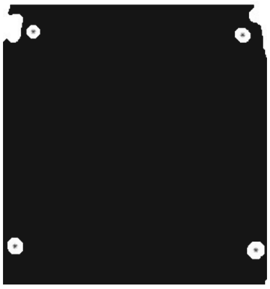


Fig. 5. Binary reference image with four landmarks centroids.

Fig. 6. Real image showing the robot and the landmark.

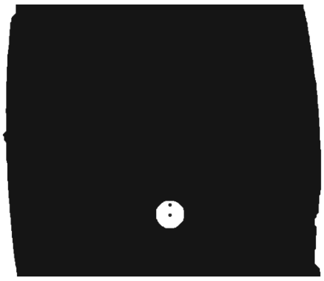


Fig. 7. Binary image showing the robot and the landmark centroids.

$$\alpha_r = tan^{-1}\frac{(Y_L - Y_R)}{(X_L - X_R)} \qquad (6)$$

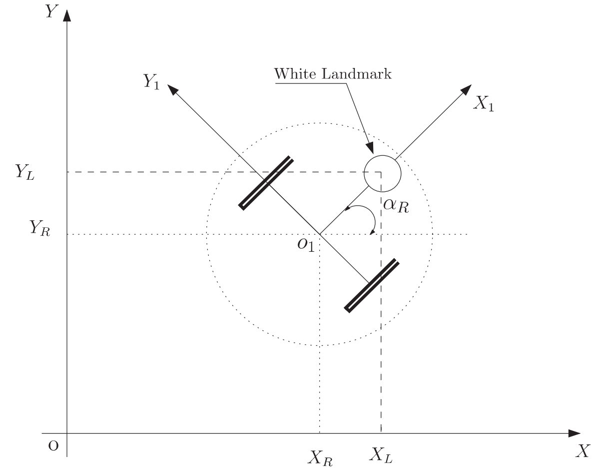


Fig. 8. Robot orientation according to the reference system.

3.3. Optimised pre-processing method

By applying the segmentation method described above on the hardware environment described by a Pentium IV PC with a frequency 3GHz, the image acquisition rate is about 5 seconds and the average processing time is about 1:44 seconds. The problem is that by using this time processing we ensure real time localization only for low velocities. So we search for decreasing this processing time. As a possible solution we apply a cropping rectangle to a binary input image in order to reduce the number of the treated pixels. The architecture of the proposed algorithm is presented in Figure 9. In the first step, we have the first binary image with the default webcam image size 640* 480 pixels (see Figure 10a). On this image we calculate the robot position in pixels coordinates as shown in step three, next we define a crop rectangle with the following form:

$$Crop_{rectangle} = [Xmin\ Ymin\ Width\ Height]$$

The two first parameters depend on the robot coordinates so that for a cropping rectangle with the size 240* 240 pixels (see Figure 10b), $Xmin$ and $Ymin$ are presented in equations 7-8.

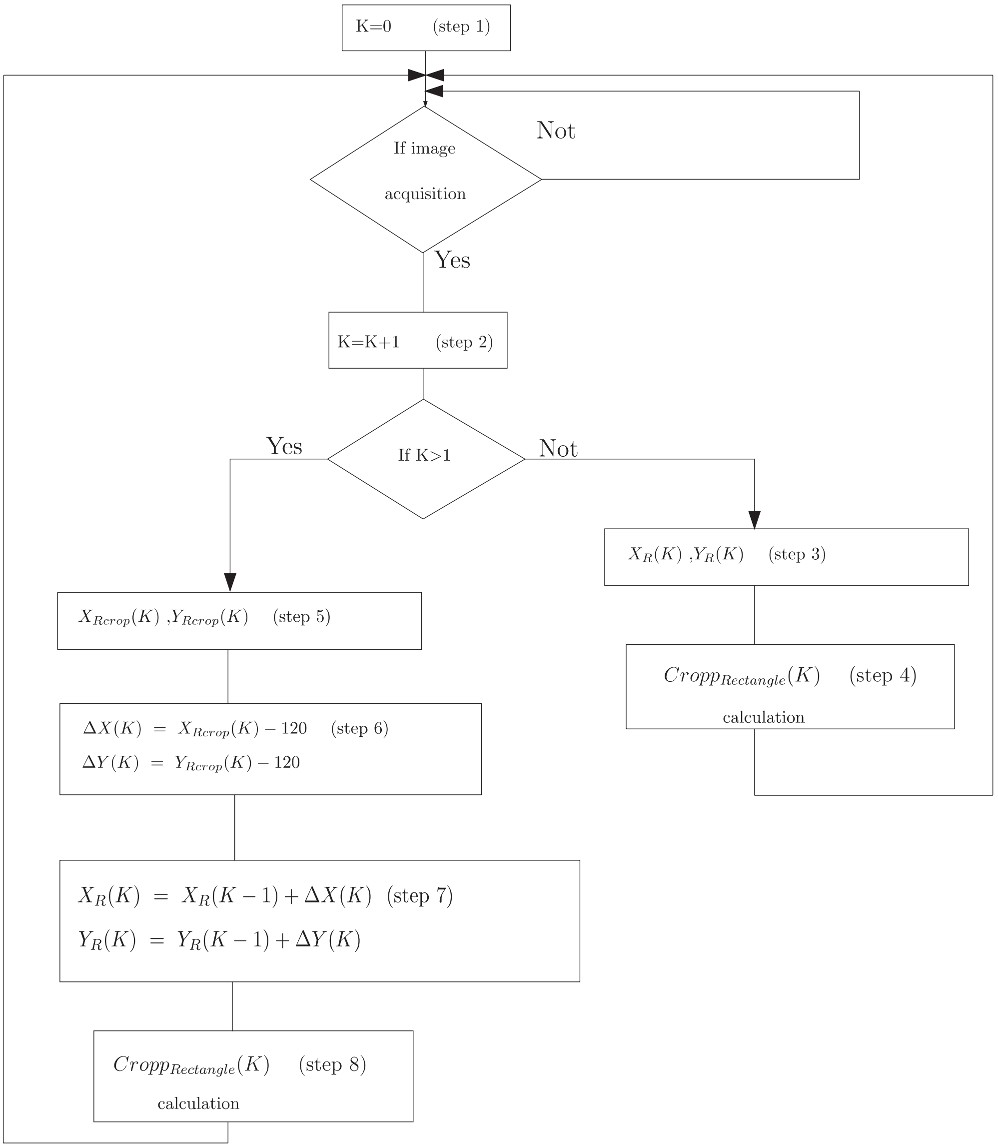


Fig. 9. Bloc diagram of the optimised pre-processing method.

*(a) First image: 640*480 pixels*

*(b) Cropped image: 240*240 pixels*

(c) Zoom of the cropped image

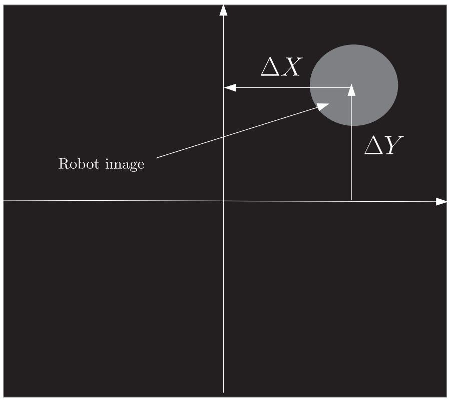


Fig. 10. Binary first image and the next cropped images.

$$Xmin = X_R - 120 \tag{7}$$

$$Ymin = Y_R - 120 \tag{8}$$

In fact, we have to note that the choice of the cropping rectangle size is based on experimental test carried out in the mobile robot so that between two following images whatever the wheels velocities, the robot reminded in the new cropped environment. Next (step five), the obtained cropping rectangle is applied on the next acquired image (see Figure 10c) and the robot position is calculated. In step six, we defined the K^{th} robot displacement beyond the $(K-1)^{th}$ robot position which is in reality the $(K-1)^{th}$ cropping rectangle centre. The step seven is used to remind the robot centroid coordinates in the reference image size defined by 640*480 pixels. The proposed optimised pre-processing method ensured an average processing time equal to 0:48 seconds and which corresponds to 70% time decrease.

3.4. Transformation from image coordinates to real coordinates

We obtain the real image coordinates by a simple difference between the pixel coordinates either of the robot or of the reference system. The resulting coordinates are multiplied by a constant coefficient K. This coefficient is calculated on the basis of both the real and the pixels distance between the landmarks along the x-axis and the y-axis. Then, we used a rule of three to transform the robot pixels coordinates to the real ones. Figure 11 describes the positioning architecture adopted.

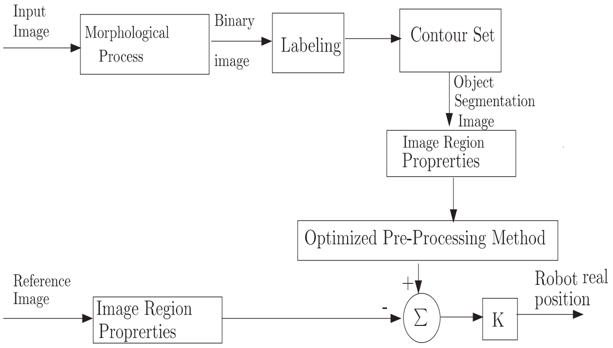


Fig. 11. Segmentation and positioning architecture.

4. Experimental Results

The proposed method in the previous sections has been implemented and tested on the mobile robot Khepera II. This robot is designed at the Swiss Federal Institute of Technology in Lausanne. It is widely used for research and teaching purposes because it allows real-world testing of algorithms developed in simulation. This Khepera II is circular of 55 mm in diameter and 30 mm in height. Its weight is only 70 g and its small size allows experiments to be performed in small work areas (see Figure 12). It has two driving wheels, which can be independently controlled. The wheel diameter is 15.2 mm. Two incremental encoders placed on each motor axis gives 12 pulses per mm. The distance between the two wheels (L) is 53 mm. This robot has a series of functions, which ensure its control either by using Matlab environment, or Lab-view environment. Two precautions to be taken during the robot command in real time. The first one is the communication speed between the robot and the host computer. The second constraint is to make sure that the wheel speed has already responded to the first instruction given to it before we could send the next one. In this work, the experimental environment used is illustrated in Figure 13. The system studied consists of a robot controlled via a serial communication from the PC. The terminal configuration for the host computer PC must be set to 57600 Bauds, 8 bit, 1 start bit, 2 stop bit and no parity.

The robot is equipped with two encoders placed on the left and right wheels and height infrared sensors. The host computer PC executes the task of calculating the optimised trajectory using the (FLC), and determining the robot relative position using odometer measurements. Besides PC is reserved to estimate the position of the robot using visual system. The communication with the webcam is ensured via USB port, with a simple protocol to acquire data. Experimental tests are carried out with the Matlab environment.

We have also developed an interface ensuring the real time robot localization with both encoders and webcam data. The interface presented in Figure 14 resumed all steps we need to control the robot. First, we search for the robot initial position and orientation by using an external sensor defined by the webcam as the encoders are unable to provide these parameters. Next we choose a virtual target to be attempted by the robot and the optimization technique to be adopted so that the fuzzy controller calculates the wheels velocities. Finally, we started the experimentation by applying these velocities to the robot and calculating the vehicle current position with encoders and webcam data.

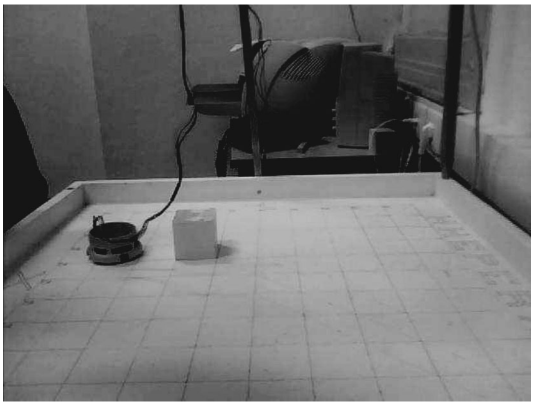


Fig. 12. Khepera II robot.

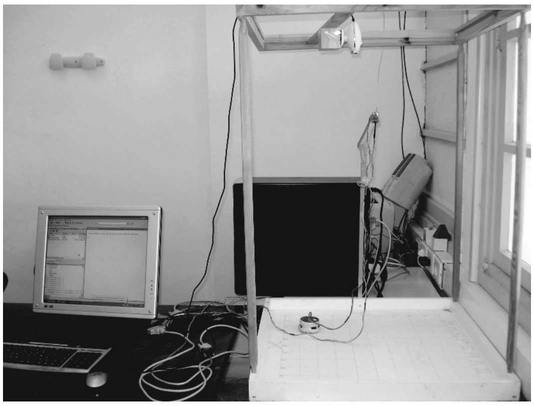
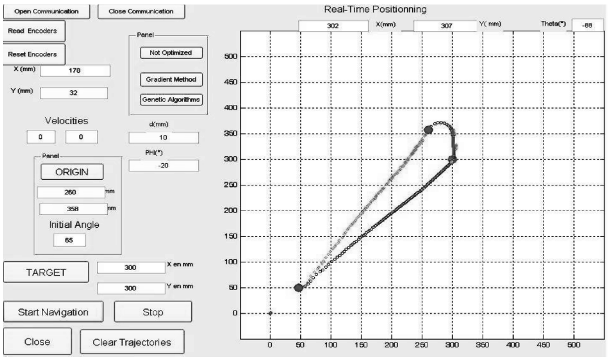


Fig. 13. Experimental environment overview.



X(mm) 302 Y(mm) 307 Theta(°) -88

X (mm) 178

Y (mm) 32

Velocities 0 0

d(mm) 10

PHI(°) -20

260 mm

358 mm

Initial Angle 65

300 X en mm

300 Y en mm

Fig. 14. Robot localization using retiming point.

4.1. Experiment 1

The purpose of the first experiment is to show the application of the segmentation method in the localization of the mobile robot with the webcam data compared to the measurement of odometers. Figures 15 and 16 show respectively the original input images and the segmented images with robot centroid's in image coordinates while the robot had to attempt a target defined by the coordinates $(x = 400\,mm; y = 50\,mm)$. Therefore, Figure 17 shows an other target defined by the coordinates $(x = 50\,mm; y = 400\,mm)$.

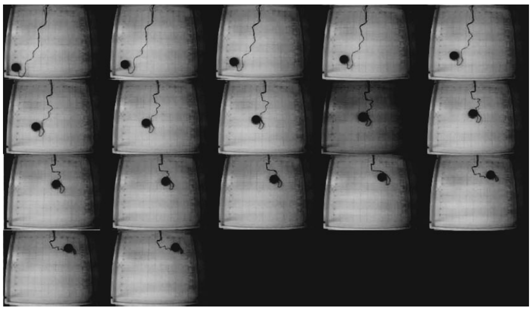


Fig. 15. 17 Frames of the original input image.

In this last experiment, there exists an approximately $2.8\,cm$ deflection along the x-axis and $0.3\,cm$ deflection along the y-axis but the encoder measurements are erroneous as shown in Figure 17. However using webcam data, the robot position and orientation become very similar to the real one. In fact, the deflection along the x-axis is very important because of the velocities generated by the fuzzy controller and applied to the mobile base are very important where the difference between initial angle and target angle is important.

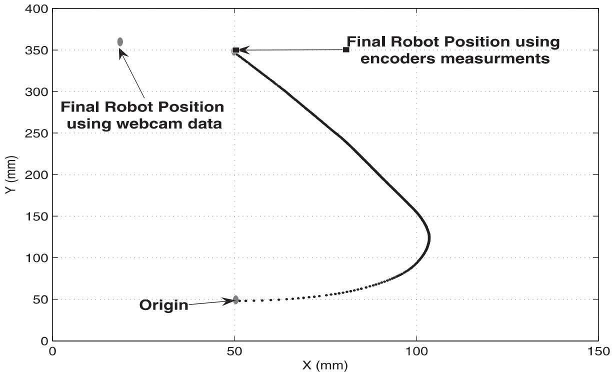


Fig. 17. Robot localization using encoder measurements.

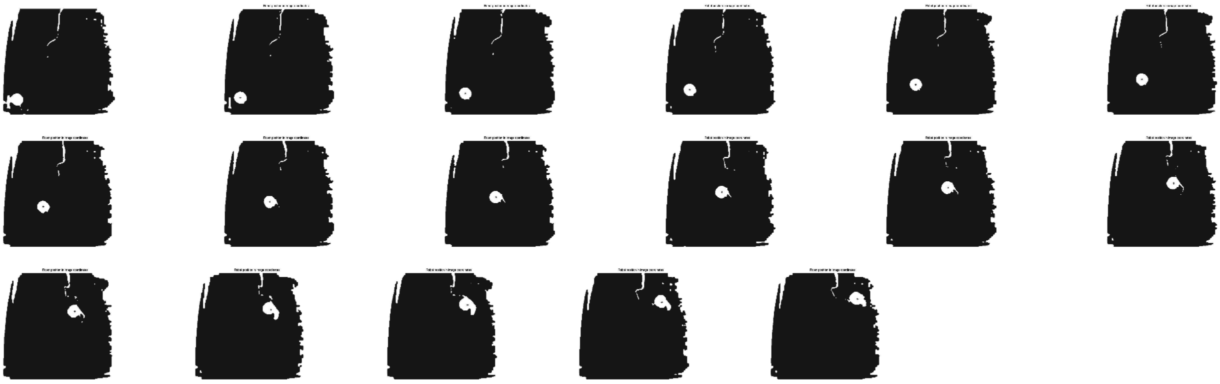
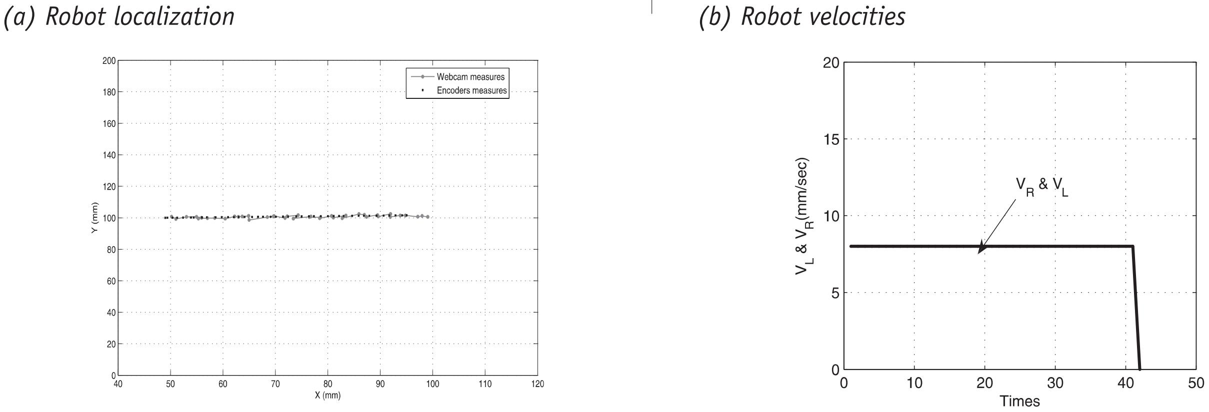


Fig. 16. Different robot positions in image coordinates (frames must be read from the left to the right).

4.2. Experiment 2

The purpose of the second experiment is to show the real time localization through both encoders and webcam data. Results of this experiment are presented in Figures 18, 19 and 21. In these experiments, the robot have to attempt targets defined respectively by $(x = 100\,mm; y = 100\,mm)$, $(x = 400\,mm; y = 100\,mm)$ and $(x = 300\,mm; y = 300\,mm)$. We notice that the error between the two measures and the deflection along x-axis and y-axis increased while both the distance and the angle moved increased. In the long trajectory experiment described by Figure 19 we have realized four real positioning robot measures (M1, M2, M3 and M4) using a ruler as shown in Figure 20. The obtained values proved that the absolute localization(visual date) is better than the odometry localization and that the webcam data are much closed to the real measurements. The results of this comparison are summarized in Table 2.

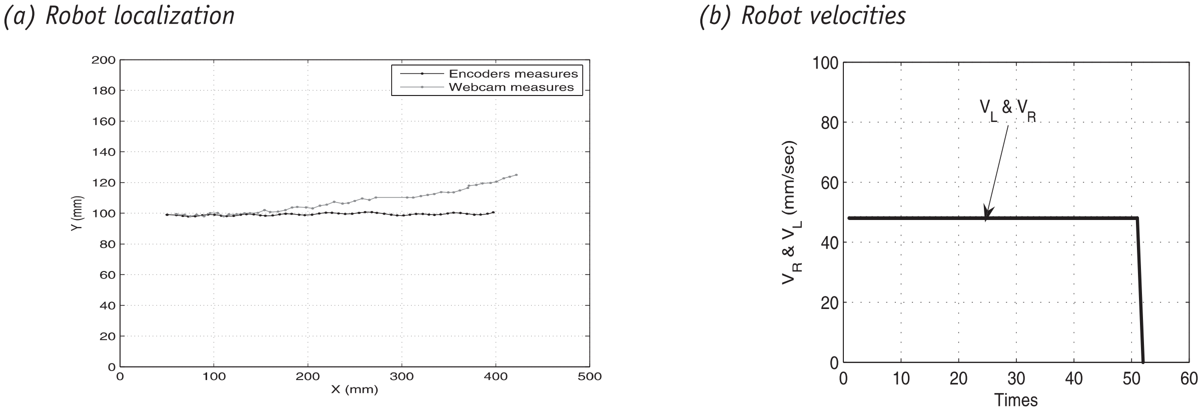
This result is due to the incremental movement of the encoders and the wheels slippage problem. So by adopting an absolute localization, the robot trajectory becomes much closed to the real one. But, to further improve the suggested method we used retiming point during the robot trajectory in order to correct the encoder measu-



(a) Robot localization

(b) Robot velocities

Fig. 18. Robot localization and velocities using both enco-ders and webcam measurement during short trajectory.



(a) Robot localization

(b) Robot velocities

Fig. 19. Robot localization and velocities using both enco-der and webcam measurements during long trajectory.

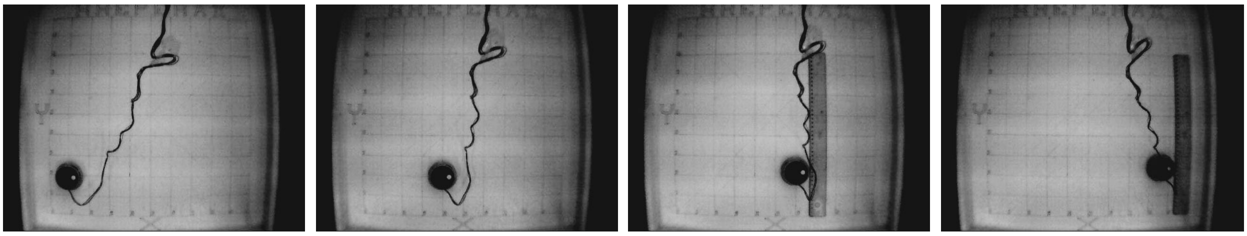
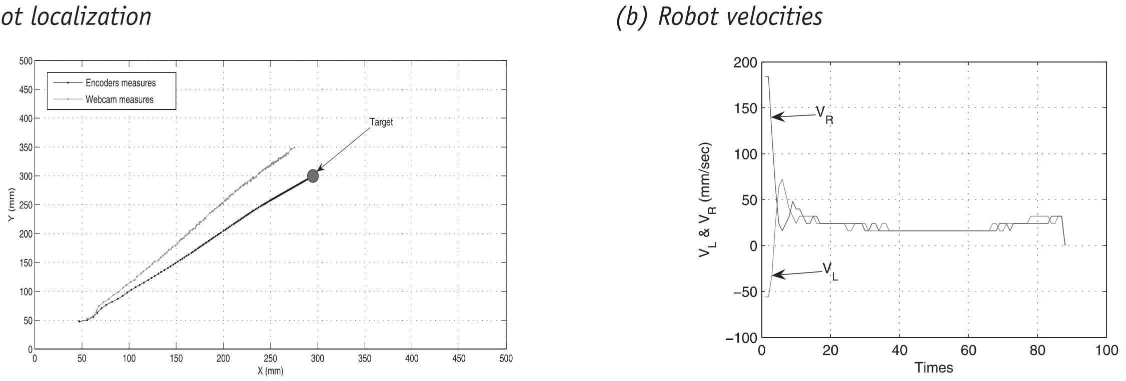


Fig. 20. Different robot real positions measured with a ruler.

Table 2. Performance evaluating table.

$X_R(mm)$ / $Y_R(mm)$		Measures							
		M1		M2		M3		M4	
		X_R	Y_R	X_R	Y_R	X_R	Y_R	X_R	Y_R
Methods	Encoders	50	99	204.4	99.06	301.9	98.57	397.4	100.6
	Webcam	50	99	198.1	103.8	305.6	110.2	421.9	124.9
	Ruler	50	100	200	104	306	115	424	125


(a) Robot localization

(b) Robot velocities

Fig. 21. Robot localization and velocities using both encoder and webcam measurements during a non-linear trajectory.

rements by the visual system measures.

4.3. Experiment 3

The purpose of the third experiment is to show the real time localization using the retiming visual control. Results of this experiment are presented in Figure 22. In this experiment, the robot has to attempt the target defined by $(x = 300\,mm; y = 300\,mm)$ from the initial position defined by the coordinates $(x = 50\,mm; y = 50\,mm)$.

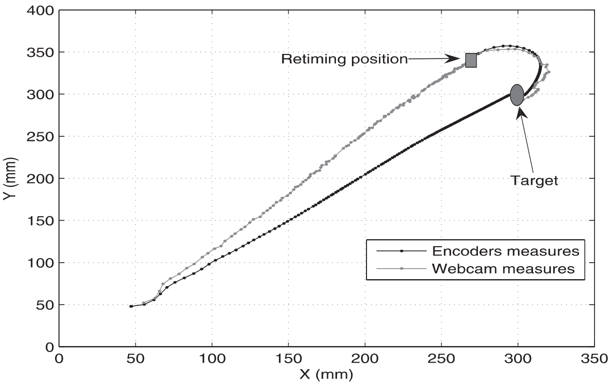


Fig. 22. Robot localization using retiming point.

We notice that with the new correction the mobile robot attempts the virtual target with a small error. In fact, we consider one retiming point at the last robot position. The calculated position by the visual system is considered as the new robot initial position used in the optimised fuzzy controller to generate the new wheels velocities in order to attempt the initial virtual target chosen at the beginning.

5. Conclusion

In this paper, we have proposed a retiming visual controller for the control of a wheeled mobile robot. Gradient method is used to optimise consequences of a Sugeno fuzzy logic optimal controller used to generate velocities to be applied on the mobile base. The experimental results effected in the Khepera II robot demonstrate that each behaviour works correctly and showed the effectiveness of this method in long and non-linear trajectories.

However, to ensure the real time localization using webcam data and because of the visual system time processing and time acquisition we have to stop the robot. This last constraint is to make sure that the wheel speed has already responded to the first instruction given to it before we could send the next one. So, the number of times we have to stop the robot depends on the command number sent needed to complete the mission(to reach the target). In fact, this constraint does not effect the estimated position neither with encoders nor with webcam. But to avoid this problem we suggested using the visual system only at few retiming point so that we correct the robot localization and we decrease the time processing. Besides and to further improve the time processing we suggested to use cropping rectangle with a dynamic size.

AUTHORS

Mohamed Jallouli*, Lobna Amouri, Nabil Derbel - Research unit on Intelligent Control, design and Optimisation of complex Systems (ICOS) University of Sfax, Sfax Engineering School, BP W, 3038 Sfax, Tunisia. Phone: (216-74) 274.088, Fax. (216-74) 275.595.
E-mails: mohamed.jallouli@enis.rnu.tn,
lobnaamourijmail@yahoo.fr.
* Corresponding author

References

[1] Thomas D. L., *Optimal Fusion of Sensors,* Department of Automation Technical University of Denmark, PHD-Essay, 1998, pp. 1-9.

[2] Green D.N., Sasiadek J.Z., and Vukovich G.S., "Path tracking obstacle avoidance and deal reckoning by an autonomous planetary rover". In: *Proc. IEEE International Conference on Robotics and Automation*, San Diego, CA, 1994, pp. 1300-1305.

[3] Carelli R., Santos-Victor J., Roberti F., Tosetti S., "Direct visual tracking control of remote cellular robots". *Robotics and Autonomous Systems Journal*, vol. 54, July, 2006, pp. 805-814.

[4] Escamilla P.J., Neil M., "Hybrid Kalman Filter-Fuzzy Logic Adaptive Multisensor Data Fusion Architectures". In: *Proc. Decesion and Control Conference*, Hawaii USA, December 2003, pp. 5215-5220.

[5] Gaone Y., Krakiwsky J., Abousalem M.A., and McLellan J.F., "Comparison and analysis of centralized, decentralized, and federated filters Navigation", *Journal of the Institute of Navigation*, vol. 40, no. 1, 1993, pp. 69-89.

[6] Sasiadek J.Z., Hartana P., "Sensor Data Fusion Using Kalman Filter". In: *Proc. Conf ISIF*, 2000, pp. 19-25.

[7] Kobayashi F., Arai F., Fukuda T., Shimojima K., Onoda M.

and Marui N., ”Sensor Fusion System using recurrent fuzzy inference”. *Journal of Intelligent and Robotic Systems*, vol. 23, 1998, 201-216.

[8] Taeseok J., Soomin P., Jang Myung L., ”A Study on Position Determination For Mobile Robot Navigation in an Indoor Environment”. In: *Proc. International Symposium on Computational Intelligence in Robotics and Automation IEEE Conf*, Kobe, Japan, 16th-20th July, 2003.

[9] Myong Ho K., Sang Cheol L., Kwae Hi L., Self Localization of Mobile Robot With Single Camera in Corridor Environment”. In: *Proc. Conf ISIE*, vol. 3, June 2001, pp. 1619-1623.

[10] Freda L., Oriolo G., “Vision-based interception of a moving target with a non-holonomic mobile robot”. *Robotics and Autonomous Systems Journal*, vol. 55, February 2007, pp. 419-432.

[11] Surmann H., Husser J., Peters L., ”A Fuzzy system for indoor mobile robot navigation”. In: *Proc. IEEE international Conference on Fuzzy Systems*, Yokohama, Japan, vol.1, 1995, pp. 83-88.

[12] Ti-Chung L., Kai-Tai S., Ching-Hung L., Ching-Cheng T., ”Tracking control of Mobile Robots Using Saturation Feedback Controller”. In: *Proc. IEEE International Conference on Robotics and Automation*, Detroit, Michigan, May 1999, pp. 2639-2644.

[13] Prahlad V., Ooi C. M., Xiao P., Tong H. L., ”Fuzzy Behavior-Based Control of Mobile Robots”. In: *Proc. IEEE Transactions on Fuzzy Systems*, vol. 12, no.4, August 2004, pp. 559-563.

[14] Cupertino F., Giordano V., Naso D., .Delfine L., ”Fuzzy Control of a Mobile Robot”. In: *Proc. IEEE Robotics and Automation Magazine*, December 2006, pp. 74-81.

[15] Hung-Ching L., Chih-Ying C., ”The Implementation of Fuzzy-Based Path Planning for Car-Like Mobile Robot”. In: *Proc. International Conference on MEMS, NANO and Smart Systems (ICMENS'05)*, 2005 IEEE, pp. 467-472.

[16] Sung-Yug C., Jang Myung L., Chung Kun S., Hyek Hwan C., ”The detection of Lanes and Obstacles in Real Time Using Optimal Moving Window”. *JSME International Journal Series - C*, vol. 44, no. 2, June 2001, pp. 567-578.