Proceedings of the
2007 IEEE Intelligent Vehicles Symposium
Istanbul, Turkey, June 13-15, 2007

Intelligent Real time Control of Mobile Robot Based on Image Processing

Nima Fatehi, Mohammad Teshnehlab, Sadaf Shariati[1]

Abstract— **Tracking control of mobile robots has many research interests among academic researchers [1],[6],[7]. This subject has opened many different aspects of research studies for different purposes such as obstacle avoidance, trajectory tracking, vision based tracking, etc. The results are being used in different autonomous vehicles from autopilot systems to little discovery mobile robots. In this paper a control scheme has been proposed for the first time to control a mobile robot using fuzzy control and image processing approaches in two cascaded loops. The image processing approach is used to estimate the traveled trajectory and configuration -defined as velocity and azimuth- of the mobile robot using special landmarks. Having appropriate feedbacks, the fuzzy controller is used to control the mobile robot at the desired configuration while traveling to the destination point.**

Keywords: Mobile Robot, Image processing, Fuzzy control, Input and output scalers, Template matching, and None-linearity

I. Introduction

In recent years the intelligent control based on neural networks, fuzzy systems, neuro-fuzzy, etc. has been a subject of widespread researches [1]. In particular scope for controlling a mobile robot in practice, intelligent control is more attractive. Intelligent control is model-free and could be easily applied to nonlinear and time variant systems [9]. A common mobile robot with two independent wheels is of such types of systems specially when navigating in an unknown environment. This paper is organized as follows; first we have made a survey on closed loop position control of a mobile robot. Then considering the robot configuration (i.e. velocity and azimuth) as a base for controlling the position we have used a *MIMO* dynamic model for mobile robot and tried to control it in a closed loop scheme using fuzzy controllers the feedback to which are provided through a camera and machine vision. Finally we have used this approach to control a common mobile robot with two independent wheels in experiment and printed out the results to demonstrate how effective the used techniques have been.

1) Manuscript received January 14, 2007. This work was supported in part by K.N.Toosi University of Technology in IRAN.

Mohammad Teshnehlab is with the Department of Electrical Engineering, K.N.Toosi University of Technology, IRAN (email: Teshnehlab@eetd.kntu.ac.ir).

Nima Fatehi is a graduated student of Electrical Engineering, K.N.Toosi University of Technology, IRAN. He is now an M.Sc Student of Azad University, IRAN (corresponding author, phone: +98-912-337-4375; e-mail: Nima.Fatehi@valiyacom.com, Neamadeus@yahoo.com).

Sadaf Shariati is a student of Electrical Engineering, K.N.Toosi University of Technology, IRAN (email: sadaf16ir@yahoo.com).

II. Control Scheme

In a closed loop control system the goal is to reach some desired parameters in the response of the system to the set point commands and unwanted disturbances [2]. When analyzing a mobile as a plant, we need to know what exactly the inputs and outputs of the system are, and what are the goals we are about to reach in the response of the output. As expected, the output of a mobile robot should be considered as its position in the environment, and the set point input would be the desired destination toward which the mobile robot has to travel.

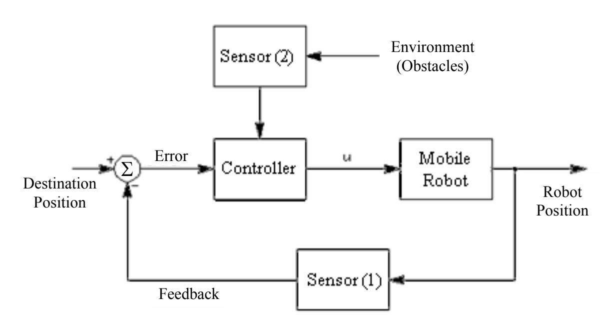


Fig. 1: Mobile robot in a closed control loop

Fig. 1 shows mobile robot in such control loop. As shown, some sensors give the controller awareness of the obstacles in the environment in order to avoid them, while some give a feedback of robot's displacement or position in order to correct the traveling trajectory. We consider a mobile robot with two independent wheels as is the case in our experimental test. We assume that the origin of the environment is the starting point of robot motion with x axis superposed on the heading of robot. Now let's consider the mobile robot in a location, **P**, between origin, **O**, and destination, **Q**, as shown in fig. 2.

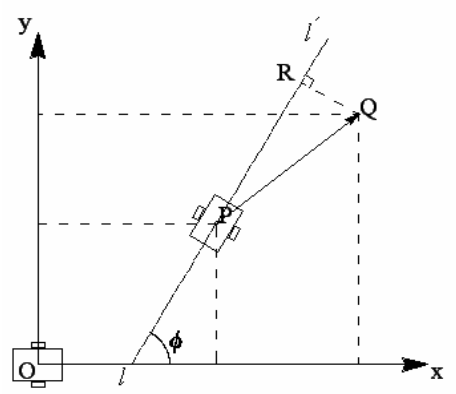


Fig. 2: Mobile robot position between origin and destination

For the mobile robot to reach the destination point Q it is necessary to change its heading (shown as line ll' with angle ϕ) to superpose on direction of vector $\overrightarrow{PQ}$. However, if this will not happen, the mobile robot can continue its motion to the point R with the minimum distance to the destination. As far as R would be, for reaching it in an appropriate time the velocity has to be higher. It could be shown [1] that a mobile robot with two independent wheels can be modeled as a MIMO system with V_l and V_r, respectively left and right DC motor terminal voltages, as inputs, and υ and ϕ, velocity and azimuth respectively, as outputs. So if we consider a velocity set point proportional to size of $\overrightarrow{PR}$ and an azimuth set point equal to $R\hat{P}Q$ this would make the best solution to get from P to Q in a minimum time and with minimum possible motion. If there would be obstacles in the environment then it should be taken into account too, but here we assume the mobile robot is moving in the x-y plane with no obstacles. Considering obstacles will change the trajectory but not the control strategy discussed. Explaining the strategy in other words, controlling the position of mobile robot is basically depended on controlling υ and ϕ which we name robot's configuration here. For this to achieve, we propose two control loops, one of which called *outer loop* for controlling the position of mobile robot considering the control strategy discussed above and the other one called *inner loop* for controlling configuration of the mobile robot. Such scheme is shown in fig. 3. Note that the feedback is considered as we will use in this approach.

III. FUZZY CONTROLLER

For controlling the mobile robot we proposed two cascaded loops named inner and outer loops. In fact the inner loop guarantees that the desired velocity and direction is reached through an appropriate time and every disturbance is eliminated by means of the inner loop controller [4]. We have used two fuzzy controllers in the inner loop, one for controlling velocity and the other for controlling the azimuth. Using fuzzy logic, as will be seen in the results, would help to apply some nonlinear characteristics to the controllers, by means of which overcoming some undesirable characteristics in the responses such as oscillation and long time delays becomes possible [5].

Modeling a mobile robot [1] we will find linear velocity, υ, of the robot proportional to average of applied inputs (i.e. voltages) to left and right motors and the angular velocity, ω, proportional to difference of these inputs. Hence we can write:

$$\begin{cases} M\dot{\upsilon} = D_r + D_l \\ I_\upsilon \dot{\omega} = D_r l - D_l l \end{cases} \tag{1}$$

where M is the mass of mobile robot, I_υ represents the inertia moment, and l is the distance between center of wheels and robot's center of gravity. D_l and D_r stand for linear forces applied by wheels to the robot chassis at left and right wings. Applying Laplace transform we can conclude from above:

$$\begin{cases} \upsilon(s) \propto (D_r + D_l) \propto \tfrac{1}{2}(V_r + V_l) = V_c \\ \omega(s) \propto (D_r - D_l) \propto \tfrac{1}{2}(V_r - V_l) = V_d \end{cases} \tag{2}$$

In (2) V_l and V_r are input voltages respectively to left and right motors, and V_c and V_d are defined as common and difference mode voltages respectively. D_r and D_l are related to motor voltages by writing dynamic equations of servo motors loaded with wheels [1].

Thus for controlling the linear velocity we use V_c and for controlling the azimuth we use V_d as control signals. Input signals to the fuzzy controllers will be error and error's time derivative. The fuzzy controller for the velocity control produces variations of V_c as output which is integrated then, while the fuzzy controller for the azimuth produces the V_d itself, because this latter signal will cause changes in ω which is integrated then in the real world to produce the azimuth, ϕ. Fig. 4 shows how these two controllers are placed in the inner loop.

For a real controller as shown in fig. 5, more than a single fuzzy controller we will need some extra parts to condition inputs and outputs of the controller to match the real world conditions. These parts include:

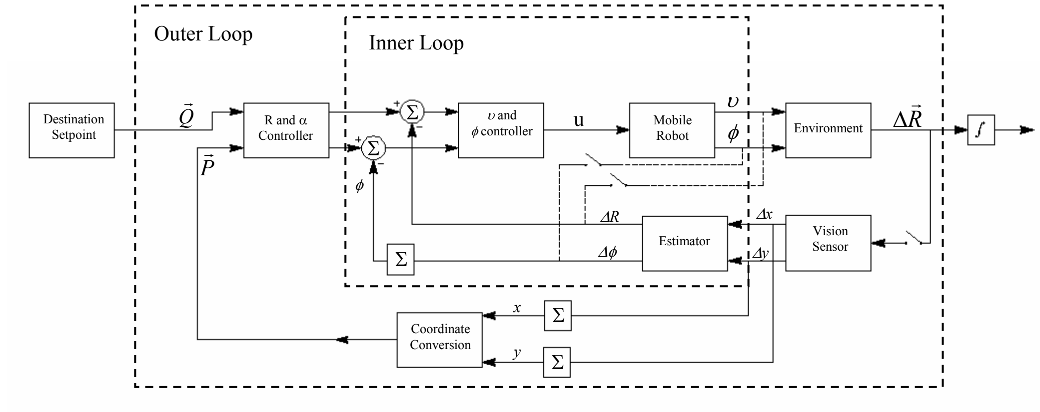


Fig. 3: Dividing mobile robot position control into two cascaded loops

- Sampler and zero order hold; since fuzzy controller is a digital controller we need sampling of input signals.
- Input and output scalers; used to map the real world values to normalized ranges in fuzzy controllers and vice versa.
- Limiters in inputs and outputs; used to limit signals which are out of defined ranges.
- Integral anti windup; used in the velocity controller due to integrator and limiter there.

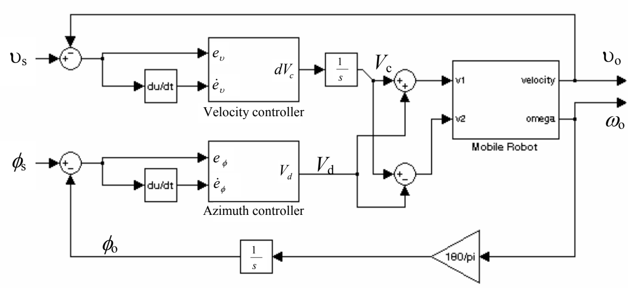


Fig. 4: Placement of fuzzy controllers in the inner loop

The next step is designing the fuzzy controllers in terms of MFs and fuzzy rules. We have used seven triangular MFs for the error inputs, five for the error derivatives and seven for outputs as shown in fig. 6. As shown in this figure, MFs are thinner around zero and grow fatter as we move toward margins. This will make a nonlinear characteristic in controller which as we may see in the results, would help to achieve improved responses. These MFs have been adjusted in term of their shapes using trial and error approach to achieve the best possible response.

The table of fuzzy rules is the same for both controllers and is shown in table 1.

Table 1: Fuzzy rules for inner loop controllers

$\dot{e}$ \ e	P	PM	PL	Z	NL	NM	N
P	*PH*	*PH*	*PM*	*PM*	*PL*	*PL*	*Z*
PL	*PH*	*PM*	*PL*	*PL*	*Z*	*Z*	*NL*
Z	*PM*	*PL*	*PL*	*Z*	*NL*	*NL*	*NM*
NL	*PL*	*Z*	*Z*	*NL*	*NL*	*NM*	*NH*
N	*Z*	*NL*	*NL*	*NM*	*NM*	*NH*	*NH*

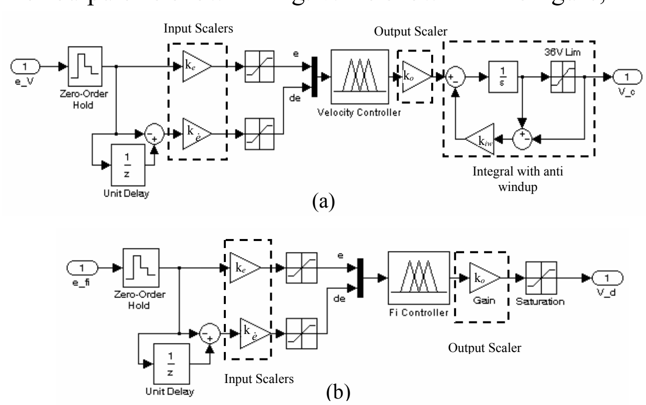


Fig.5: Conditioning signals for interconnection of inner loop controllers; (a) velocity controller, (b) azimuth controller

IV. VISION FEEDBACK

A camera is fixed on the robot chassis which looks down at the ground to see how the environment is moving. Since robot motion and hence camera motion relative to the ground is proportional to the seen image motion relative to the camera, we find the motion of the seen image in the camera between two sequential frames in terms of Δx and Δy and then our feedback parameters, robot displacement and heading angle, could be found from Δx and Δy.

We have used template matching based on correlation of images to find displacement of two sequential images [3]. Also we have used some artificial landmarks to make this approach more accurate and easy. The template is selected as a central window from frame F_n and is compared to match somewhere in the frame F_{n-1}. The matching coordinate (x and y) is found as the maximum of the

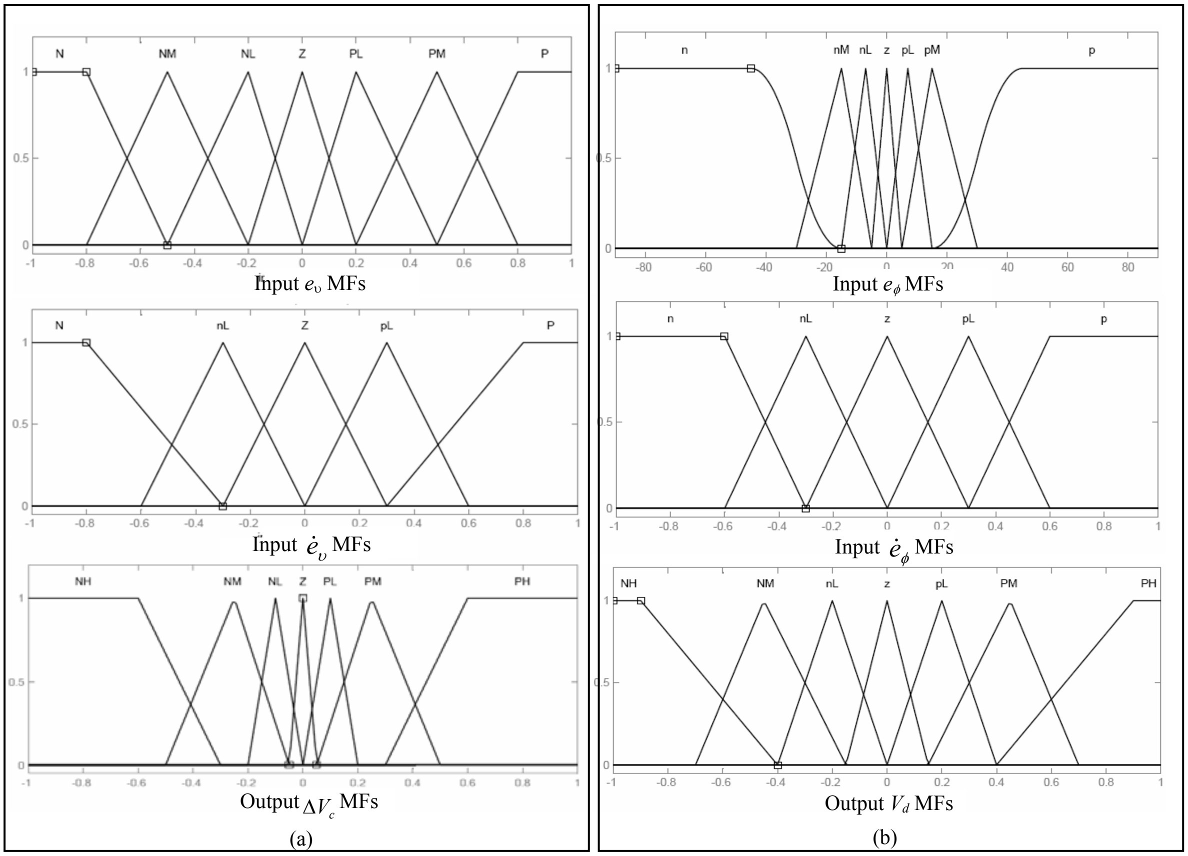


Fig. 6: Shape of membership functions; (a) velocity controller, (b) azimuth controller

correlation function between zero padded template(i.e. central window) and the frame F_{n-1}. The correlation is performed in the frequency domain to decrease the calculation time. Fig. 7 shows the image matching results.

In the image conditioning section features are extracted and some unwanted effects of the images such as geometrical distortion are improved. Then a comparison will be made in the image matching section between central window of the current frame and whole of the previous one. The result will be in terms of *x* and *y* which then will be converted as the displacement of robot in terms of ΔR and $\Delta\phi$. In fig. 8 the rectangle C corresponds to the frame seen through the camera. We can see from fig. 8 that:

$$\Delta\phi = \tan^{-1}\left(\frac{\Delta x}{kl}\right) = \tan^{-1}\left(\frac{\Delta x}{k_x}\right) \qquad (3)$$
$$\Delta R = k_y \Delta y$$

Where k_x and k_y are constants for calibration. These constants are found by assigning a linear calibration equation between the actual displacement/rotation of mobile robot and calculated $\Delta\phi$ and ΔR from image processing results.

By summing ΔRs and $\Delta\phi$s we will find the final location of mobile robot as the feedback to our controlling loop.

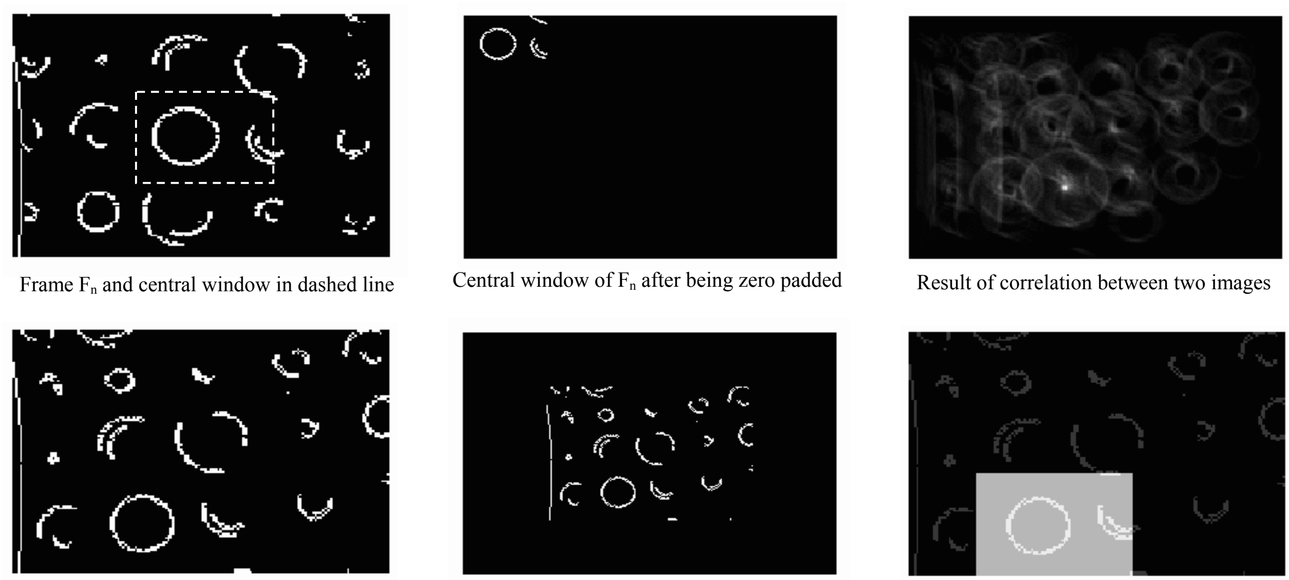
Here we have used such feedback system; however the displacement and heading angle feedbacks could be achieved using many other sensors such as shaft encoders and range scanners [8].

V. Experiment Results and Tuning

Utilizing the designed controller and vision feedbacks as described above, we have brought here the final results of the inner loop responses. The goal has been to reach the desired set points in an appropriate time and with minimum oscillations. Long time delays and oscillations would occur in the responses of system especially due to nonlinear properties in the mechanical systems of robot the most important of which is a dead zone characteristic. This characteristic is caused by backlash and friction in gearboxes used for coupling motors to the wheels [5].

At each sample time feedback should be caught and control signals should be estimated and applied to the system. So a timer is used in the controller code which performs the jobs in the block diagram of fig. 9 every sample time. Set points are got from user in the GUI[1] of the

[1] Graphical User Interface



Frame F_n and central window in dashed line

Central window of F_n after being zero padded

Result of correlation between two images

Frame F_{n-1}

Frame F_{n-1} with zeros padded to its border

The matching place

Fig. 7: Matching two consequent images to find displacement

computer software and feedbacks are got from the image processing section mentioned above. By comparing these two, errors are found and error rate is estimated using time difference and some digital filtering for eliminating noise. These signals then will be input to the controllers, *V-Control* which stands for velocity controller and *Fi-Control* which stands for azimuth controller. The controllers are realized as fig. 5 with a Matlab code. The outputs of controllers, V_c and V_d as discussed in section 3, are then converted to bytes in order to be applied to the real system through serial communication. An electronic circuit implementing an AVR microcontroller and some drivers then uses the serial data to drive left and right motors proportional to the received control signals.

One important subject to be considered is tuning the scaling gains for inputs and outputs in the controllers. The input scaling gains are chosen so that the incoming signals would be mapped on whole the normal range of the fuzzy controller (i.e. [-1 1]). The output scaling gains on the other hand are chosen with trial and error to reach some reasonable response. High output gains would produce oscillations in the output whiles low gains would cause long time delays due to the dead-zone characteristic of the electro-mechanical system. Fig.11 shows results of changing scaling output as the gain of controller.

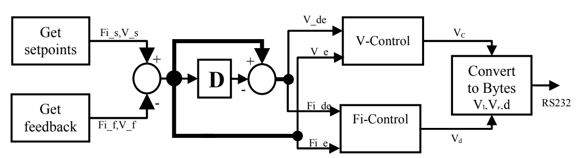


Fig.9: Block diagram of jobs in one sample time for closed loop

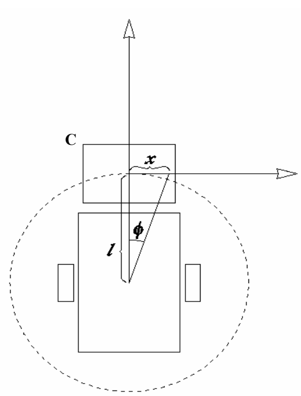


Fig. 8: Relation of image displacement in camera with robot motion

Using nonlinear characteristics for the controllers, as mentioned in the section three by choosing narrower MFs near zero and fat ones in the margins helps to avoid some undesirable effects in the outputs due to dead zone characteristic. Using such nonlinearity in the velocity controller eliminates the large starting delay, while in the azimuth controller it helps increasing response speed while eliminating oscillations. In our experiment results we have compared a response to set points of $\upsilon_{sp}=0.18^{m/s}$ and $\phi_{sp}=90^{deg}$ for a linearly designed fuzzy controller (i.e. MFs designed with similar widths so that properties of a linear controller is achieved) and the nonlinear one we discussed in section three. The results are shown in fig. 11. As can be seen in fig. 11, for we have chosen a large controller gain in order to eliminate long period delay in the output, oscillations have been generated in the output. These oscillations are undesirable specially when the response is reaching the desired value. It means oscillation of the mobile robot's heading which brings it to a zigzag motion.

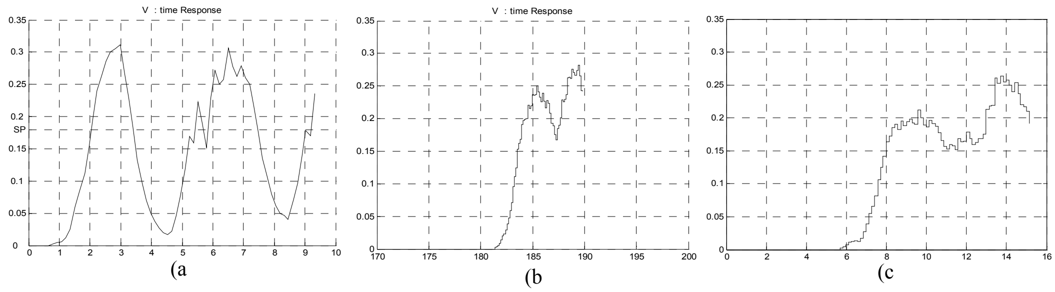


Fig. 10: Effects of changing the output scaling gain for velocity response
(a) k=12; Large oscillations, (b) k=0.1; 180^{sec} delay, (c) k=2; a reasonable response

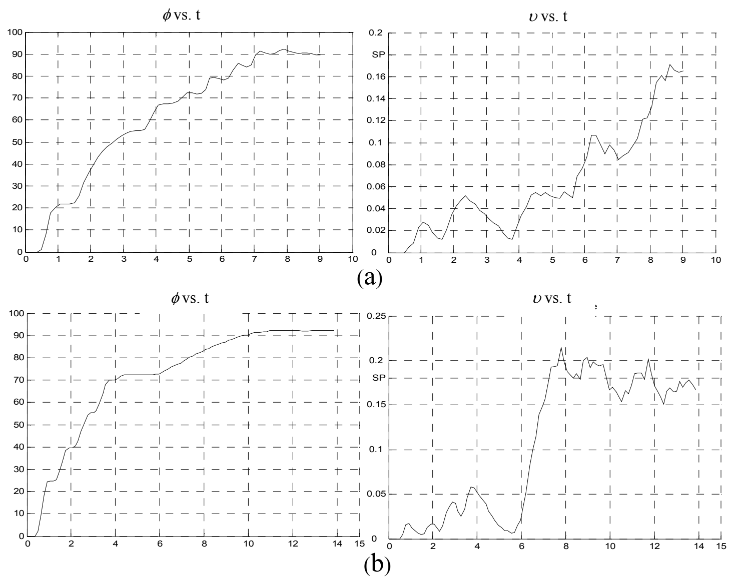


Fig. 11: Effects of using nonlinear properties in the fuzzy controller
(a)Responses by linear controller, (b) Responses by nonlinear one

On the other hand by implementing the nonlinear controller we will see oscillations during 4 seconds when the error is too high, but when the error is descended to about 30^{deg}, the controller will drive the output slowly to reach the desired value and this is when the velocity increases to reach the desired value as well.

VI. Conclusion

In this paper we have proposed a closed loop system consisting two cascaded loops (inner and outer) for controlling the position of a mobile robot. Then for realization of the inner loop we used fuzzy controllers with nonlinear characteristics and showed that these controllers can perform better than linear ones for eliminating undesired characteristics in responses rather than similar linear ones. We also used vision as a feedback of displacement to be used for the controlling purpose. The next step will be designing and implementing the outer loop to control the actual position of the mobile robot having velocity and azimuth in hands.

References

[1] K. Watanabe, J. Tang, M. Nakamura, S. Koga, T. Fukuda: *Mobile Robot Control Using Fuzzy-Gaussian Neural Networks*, Proceedings of the 1993 IEEE/RSJ International Conference on Intelligent Robots and Systems Yokohama, Japan July 2630,1993.

[2] K. Ogata**:** *Modern Control Engineering (4th Edition)*, copyright 2002 by Pearson Education, Inc.

[3] Rafael C. Gonzalez, Richard E. Woods: *Digital Image Processing (2nd Edition)*, Copyright 2002 by Prentice-Hall, Inc.

[4] Karl Astrom, Tore Hagglund : *PID Controllers, 2nd Edition*, Copyright Instrument Society of America 1995.

[5] Panos J. Antsaklis, Kevin M. Passino, eds.: *An Introduction to Intelligent and Autonomous Control*: Kluwer Academic Publishers, Norwell, MA, 1993. Available: http://www.ece.osu.edu/~passino/books.html.

[6] Michail G. Lagoudakis, Anthony S. Maida: *Neural Maps for Mobile Robot Navigation,* International Joint Conference on Neural Networks, 1999. IJCNN '99, vol.3, pp. 2011-2016.

[7] Guilherme N. DeSouza, Avinash C. Kak: *Vision for Mobile Robot Navigation: A Survey*, IEEE transactions on pattern analysis and machine intelligence, vol. 24, no. 2, February 2002.

[8] Jonathan Dixon, Oliver Henlich: *Mobile Robot Navigation*. Available: http://www.homes.doc.ic.ac.uk/~nd/surprise_97/journal/vol4/jmd/.

[9] J. Yen and R. Langari: Fuzzy Logic Intelligence, Control, and Information. Prentice Hall, 1999.