Rôbốt di động bám quỹ đạo tự động dựa trên xử lý ảnh sử dụng bộ điều khiển lai PID-mờ

Vision Based Autonomous Path/Line Following of a Mobile Robot Using a Hybrid Fuzzy PID Controller

Pham Xuan Thuy
Le Quy Don Technical University
e-Mail: thuy.phxuan@gmail.com

Nguyen Tang Cuong
Le Quy Don Technical University
e-Mail: ntcuong210@gmail.com

Tóm tắt
Trong bài báo này, chúng tôi đề xuất một phương pháp điều khiển cho robốt tự hành để bám quỹ đạo cho trước trong môi trường trong nhà. Để bám được theo quỹ đạo, các kỹ thuật sử dụng logíc mờ với thông tin đầu vào từ hệ thống xử lý ảnh đã được nghiên cứu và áp dụng. Phương pháp điều khiển sử dụng trong hệ thống là sự kết hợp giữa bộ điều khiển lái – bộ điều khiển lai PID-mờ, và bộ điều khiển vận tốc tuyến tính – bộ điều khiển mờ. Để khẳng định tính đúng đắn của giải thuật điều khiển, chúng tôi đã tiến hành thử nghiệm trên một robốt tự hành, robốt thiết kế dựa trên cơ sở các linh kiện sẵn có trên thị trường. Thử nghiệm chỉ ra rằng, robốt sử dụng giải thuật chúng tôi đề xuất bám tốt theo quỹ đạo cho trước – bao gồm cả quỹ đạo cong và quỹ đạo thẳng.
Từ khóa: Hệ thống điều khiển, robốt tự hành, hệ thống xử lý ảnh, logíc mờ, bộ điều khiển lái, bộ điều khiển tốc độ.

Abstract
In this work, we propose a method for autonomous path/line following of differential-drive wheeled robot to track straight and curved paths in an indoor environment. In order to follow the path/line, fuzzy logic techniques are applied to generate movement with the information extracted from vision system. The control method in the system combines a steering controller, which is implemented by an intelligent hybrid fuzzy PID controller, controlling the steering angle of the robot, and fuzzy velocity controller, which controls the forward linear velocity with the purpose of safe tracking. We do experiments using an open source differential-drive wheeled robot platform (Figure 1). In the experiments, the robot is successfully able to follow a pre-defined path/line containing both straight and curved sections. The effectiveness of the control algorithm, combining steering and velocity controllers, is examined in these experiments.
Keywords: Control systems, vision system, fuzzy logic, steering controller, velocity controller.

List of Symbols

Symbols	Unit	Meaning
L	m	distance between the two driving wheels
R	m	radius of the wheel
d	m	distance between the wheel´s axis and the central of ROI
θ	rad	angle between the direction of the robot and the x axis
V_l, V_r, V	m/s	the left, right wheel velocities and linear velocity of the robot respectively
E	m	distance between the directive axis of the robot and the path in ROI
ω_l, ω_r, ω	rad/s	angular velocities of the two driving wheels and the mobile robot

Abbreviation

ROI	region of interest
IPM	inverse perspective mapping
RMS	root mean square

1. Introduction
Autonomous mobile robots are machines, which are characterized by nonlinear and highly complex dynamics. Research on mobile robots is very broard because of their ability to work through large application domains, such as: (1) transportation, (2) planetary exploration, (3) surveillance, (4) security, (5) military targets tracking, and (6) human-machine-interfaces for people with mobility deficiency, and

Figure 1 *Wheeled-drive mobile robot prototype*

has still many open problems. One of the critical problems is path-tracking robots, which includes a variety of theories and technologies such as ultrasonic mapping and vision systems.

Indeed, the process that deals with this complex problem involved handles many issues including acquisition and processing of sensory data, decision making, trajectory planning, and motion control [1]. In addition, it is difficult to accurately model the interaction between the ground and the robot. Moreover, a number of subsystems in mobile robots exhibit nonlinear and time-varying behavior [2].

However, there are several solutions from the motion control point of view that have been proposed and applied in recent years. Firstly, the adaptive method [3, 4] can adjust control law by the variable parameters on system, and a certain performance criteria could be achieved. By contrast, its realization is complex, and it is not easy to meet the real-time performance of mobile robot motion control. Secondly, the backstepping method [5, 6] is widely used for tracking, but the controller structure, design and implementation process are complicated. In addition, it requests the robot to be able to give as large acceleration as possible, which may be unsafe for robot movement. The other one that is sliding mode control [7-9]. This one has outstanding merit of robustness against structured and unstructured uncertainties, on the other hand, exploring uncertainty and determining the information about the uncertainty is difficult in practice.

Intelligent techniques provide a promising approach to solve the issue of motion control with capacity of no longer relying on mathematical models and getting out of the linear constraints. In [1, 2, 10-12], the authors proposed fuzzy logic method to overcome nonlinearities, and uncertainties of the system model. However, fuzzy controllers cannot be totally summarized for influence by the subjective factors of a person, in addition, the steady state error is hard to be eliminated, and this affects the quality of control. The optimization algorithms, which do not need to consider the dynamic equations and have excellent characteristics, must be good and while hardware platform is strong enough so that convergence speed keeps up with the motion of the robot [13-15].

In this paper, an intelligent hybrid fuzzy PID controller for steering purpose and fuzzy controller for efficient velocity of robot are introduced. The hybrid controller is developed to deal with the friction disturbances, gear backlash, wheel slippage, and modeling uncertainties and the fuzzy controller is used to provide suitable velocity when mobile robot moves in different form of paths. The input of the system, which comes from vision system, will be carefully examined with image processing techniques. As a result, a higher path/line tracking performance of the wheeled-drive mobile robot can be achieved. Comparative experimental results will be presented to verify the effectiveness of the proposed control strategy in actual implementation.

This paper, consisting of six sections, can be organized as follows. Section 2 presents the kinematic model of the mobile robot. Vision system, which provides the path/line information, with effectively proposed image processing techniques are described in section 3. Section 4 illustrates the proposed control scheme, which emphasizes the reasoning methods for constructing a hybrid steering fuzzy PID controller and a velocity fuzzy controller. Experimental results of the proposed algorithm with a mobile robot are given in section 5, followed by the conclusion in section 6.

2. Kinematic model of mobile robot

The mobile robot used in this study has two driving wheels at the back, powered independently and a free wheeling caster to support the mobile platform at the front of the robot. Fig. 2 indicates a sketch of top view of the mobile robot and all relevant parameters of the robot. The mobile robot can steer towards the left or the right by differentially changing V_l and V_r.

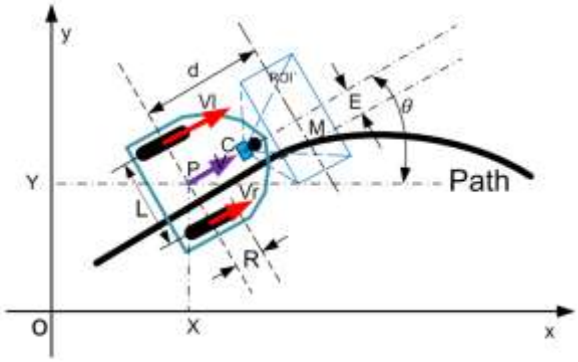

Figure 2 *Top view of the mobile robot*

It is assumed that the masses, which have the center in the middle of axis connecting the centers of the two differential wheels, and the inertia of the wheels are negligible. In addition, the wheels are non-deformable and roll without lateral sliding. At the point P(x, y), the kinematic equations describing the model of the mobile robot are given as follows [15, 16]:

$$\begin{cases} V_l = R\omega_l \\ V_r = R\omega_r \end{cases} \tag{1}$$

$$\begin{cases} \dot{x} = V\cos\theta \\ \dot{y} = V\sin\theta \\ \dot{\theta} = \omega \end{cases} \tag{2}$$

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} \frac{R\cos\theta}{2} & \frac{R\cos\theta}{2} \\ \frac{R\sin\theta}{2} & \frac{R\sin\theta}{2} \\ \frac{R}{L} & -\frac{R}{L} \end{bmatrix} \begin{bmatrix} \omega_l \\ \omega_r \end{bmatrix} \tag{3}$$

As a result, the robot can be controlled by the parameters ω_l and ω_r, which are the angular velocities of the wheels of the robot.

3. Vision System

The vision system acts as a smart sensor that acquires and processes the information needed to control the robot along the path/line. More specifically, after extracting the path/line on the ground from images captured by a pinhole camera and creating a scanline, the vision system determines the parameter E (Fig. 1), which is distance between the directive axis of the robot and the center of the path. This parameter is used as input into our proposed algorithm.

To accomplish this task, a USB camera is mounted on the mobile robot at fixed position – point C in Fig. 1. The image processing algorithm, implemented by LabVIEW programming language, has a simple structure, with a modest consumable time – from 8,5 to 10 frames/second, represented in Figure 3.

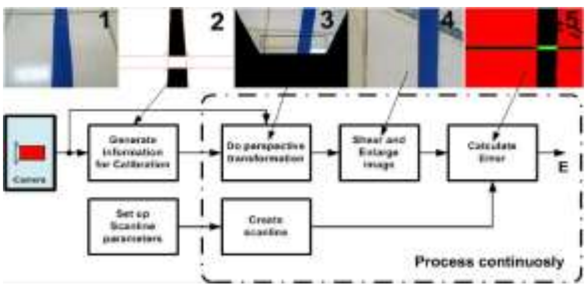

Figure 3 *Block diagram of image processing*

3.1 Calibration information

In our study case, path/line boundaries are parallel to each other, but in the 2D perspective view, they seem to converge to a point, the vanishing point. As a result, there is the shape distortion in the path/line boundaries image captured by the camera in the 2D perspective view. There are several methods proposed to cope with this problem [17, 18]. However, those methods are quite sophisticated and are time consuming. We provide a method that is simple enough in order to run fast on a software or hardware implementation. The main idea behind our way is to exploit the information contained in the path/line instead of designing a specific calibration grid or elemental patterns. To do that, firstly, a straight path/line is set up on the ground. Then, the robot, which has a pinhole camera with fixed position mounting on, is placed on the ground so that images acquired have that path/line in the center of frames as picture (1) in Fig. 3. After that, four reference points, which is necessary to do calibration [18] and used as the inputs for the perspective transformation process, with absolute 2D coordinate values are generated as picture (2) in Fig. 3. In fact, to achieve a precise calibrating result, many corresponding points should be evenly laid out across the entire image. However, the process needs to be as simple as possible to have quickly response for the robot system. The simple algorithm for creating four reference points is represented in Figure 4. To set the calibration information, we used LabVIEW IMAQ Learn Calibration Template VI and IMAQ Set Calibration Info VI, which are described in [19].

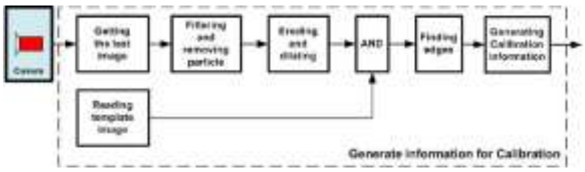

Figure 4 *Block diagram of creating calibration information*

3.2 Inverse perspective mapping

Inverse perspective mapping [20-23] is a geometrical transformation technique whereby a coordinate system can be transformed from one perspective to another. Mathematically, IPM can be described as a projection from a 3D Euclidean space, W = {(x, y, z)}∈E^3 (real world space), onto a 2D plannar, I = {(u, v)}∈E^2 (remapped image space). Figure 5 shows the relationships between the two spaces, W and I. In order to generate a 2D view of 3D scene, it is assumed that the ground is flat and the following parameters are known:

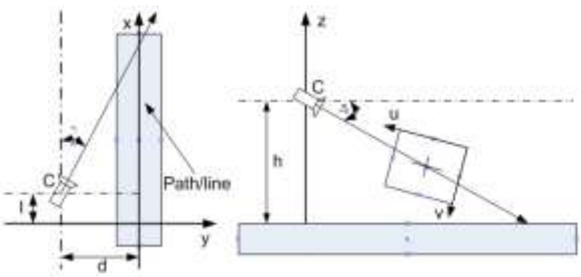

Figure 5 *Relation between world plane and image plane*

- the camera position, C = (l, d, h) ∈ W
- the optical axis of the camera: the yaw angle (γ) and the inclination angle (β)
- the camera angular aperture 2α
- the camera resolution is $m \times n$

According to [23], the transformation: (W $\longrightarrow$ I) is given as follows:

$$\begin{cases} u(x,y,0) = \dfrac{\beta(x,y,0) - (\beta-\alpha)}{2\alpha} * (n-1) \\ v(x,y,0) = \dfrac{\gamma(x,y,0) - (\gamma-\alpha)}{2\alpha} * (m-1) \end{cases} \quad (4)$$

Where

$$\begin{cases} \beta(x,y,0) = arctg\left[\dfrac{y-d}{x-l}\right] \\ \gamma(x,y,0) = arctg\left[\dfrac{h\sin\beta(x,y,0)}{y-d}\right] \end{cases} \quad (5)$$

While the transformation: (I $\longrightarrow$ W), (x, y, z), can be obtained by:

$$\begin{pmatrix} h\cot\left[(\gamma-\alpha) + u\dfrac{2\alpha}{m-1}\right]\cos\left[(\beta-\alpha) + v\dfrac{2\alpha}{n-1}\right] + l \\ h\cot\left[(\gamma-\alpha) + u\dfrac{2\alpha}{m-1}\right]\sin\left[(\beta-\alpha) + v\dfrac{2\alpha}{n-1}\right] + d \\ 0 \end{pmatrix} \quad (6)$$

In practice, there are some factors that need to be considered when implement IPM to achieve good transformation. First, this is the position of the camera. For simplistic and precise purposes, the camera viewfinder should be laid in the center of the image and focusing the vanishing point. In addition, setting up the inclination angle (β) is also crucial. If it is too narrow, the remapped image will covers too much of the ground surface and produces unwanted information. And if the inclination angle (β) is too wide, only a small area will be captured. This maybe causes a big problem for the motion control of the robot.

In the our system, IPM is implemented by IMAQ Correct Calibrated Image VI in which the inputs of the function are a new image acquired and the calibration information from the previous step. The picture (3) in Fig. 3 is an example of this transformation.

3.3 Generating vision system output

The purpose of adopting of an image processing algorithm in the robot is to provide good information about the path/line on the ground. There are several approaches that determine which information is. In [1, 14, 24], the authors estimated the parameters such as the value of curvature in the bend ahead of the mobile, the lateral error and the heading error. However, with an unknown path/line, the estimation of the value of curvature is very difficult and time consuming. In [25, 26], the authors proposed the luminance and intensity difference between the left half and the right half of the image, respectively. They are simple, but do not tell enough information about the path/line near around the robot.

We propose the parameter E, which provides information about how the mobile robot moves with respect to the path/line, as indicated in Fig. 2. This parameter can be calculated by first discarding the non-information part of the calibrated image, the shearing, – the lower part of the picture (3) in Fig. 3. After that, a ROI, which is selected by trial and error method, is extracted, and then enlarged – picture (4) in Fig. 3. Then, the path/line in the enlarged image is extracted by common image processing techniques such as filtering, removing particle, eroding and dilating. Simultaneously, a scan-line, which includes even elements, is created. Finally, the result image is added with the scan-line. The position of a scan-line is at the center of the image, and the number of its elements in the left and the right half of the image is the same. In addition, all elements fit into the image, and only the elements of the path/line position have the value 1, others are 0 – Fig. 6. As a result, the parameter E is calculated.

Figure 6 *Calculation of the parameter E*

4. The proposed control scheme

We have structured the automatic robot control system in a combining architecture of the steering controller and the velocity controller. The block diagram representing the proposed control scheme is given in Figure 7. The hybrid steering controller, which has an output U_s, controls the steering angle of the mobile robot. The reason of using the hybrid architecture is to take the advantages of both the fuzzy controllers and the PID controllers. As stated above, the robot system has lots of non-linearities. The fuzzy controllers, which emulate the human behavior, do not require an accurate mathematical model and have a quick response and that means they are good with nonlinear systems. However, the PID controllers do not deal with non-linearities well, but are more accurate and precise than the fuzzy controllers. To decide which controller to be used, a selector is constructed. The velocity controller with the output U_v, on the other hands, controls the forward linear velocity of the mobile robot for safe tracking of a continuous path/line on the flat ground.

It is noted that the two outputs, U_s and U_v, are reference inputs to the low level driver of the robot which regulates the left and the right wheel velocities. The parameters V_l and V_r can be obtained by the following

$$\begin{cases} V_l = U_v + U_s \\ V_r = U_v - U_s \end{cases} \quad (7)$$

Next, the controllers are described in details.

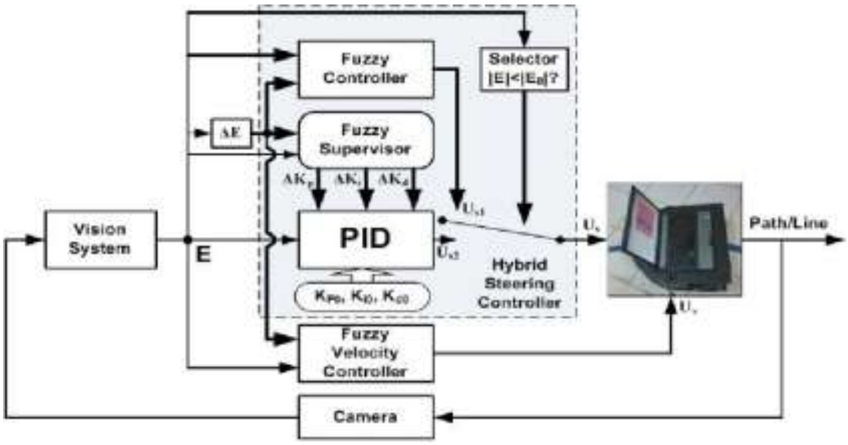

Figure 7 *The block structure of the proposed control scheme*

4.1 The fuzzy steering controller

The objective of this controller is to provide the controlling signal, U_{s1}, so that the distance error, the parameter E, is quickly minimized. Using the error, we define the two linguistic variables for the fuzzy controller as E and change of E, ΔE. The fuzzy controller structure is shown in Figure 8 [27].

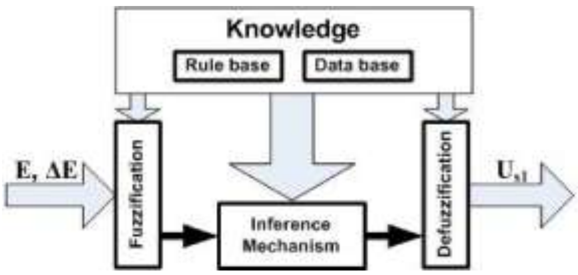

Figure 8 *The fuzzy steering controller*

The roles of each block are the following:

- the fuzzification converts the input crisp values (E, ΔE) into linguistic terms of the input variables, with a correspondent certainty value.
- the knowledge stores the data that defines the input and the output fuzzy sets, as well as the fuzzy rules.
- The inference mechanism applies the fuzzy rules to the input fuzzy variables to obtain the output variable.
- the defuzzification achieves crisp output signal, U_{s1}, based on the output fuzzy sets obtained as the result of fuzzy reasoning.

In our system, the input and output fuzzy variables (E, ΔE, and U_{s1}) have a set of seven linguistic terms {NB, NM, NS, ZE, PS, PM, PB}. The membership functions of these variables are defined of triangle shapes because of its simplification and effectiveness [28] as represented in Figure 9, 10.

The fuzzy rules are constructed based on the robot itself and the expert experience, which is actually a cut and trial procedure. This procedure is performed based on the fact that is as follows.

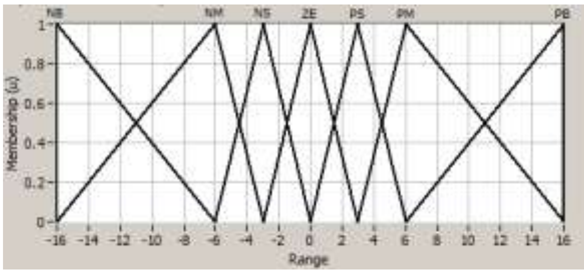

Figure 9 *The membership functions of (E, ΔE) variables*

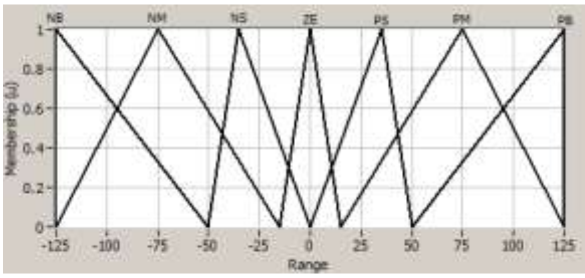

Figure 10 *The membership functions of U_{s1} variable*

If E and ΔE are both positive, it means that the path/line is to the right of the robot and the robot is moving further towards the left away from the path/line. As a result, a positive value of U_{s1} needs to provide to the robot so that the robot moves towards the right to follow the path/line. Similarly, if E and ΔE are both negative, in other words, the path/line is to the left of the robot and the robot is moving further away from the path towards the right. In this situation, a negative value of U_{s1} is necessary for the robot in order to the robot can track the path/line. On the other hands, when E is positive and ΔE is negative, the robot is in the process of moving towards the path/line. It is the same for the case with E is negative and ΔE is positive. Depending on how large values of E and ΔE are, the value of the output linguistic variable, U_{s1}, can be chosen. The justification of fuzzy control rules based on "scale mappings" method as shown in [29]. As a result, the fuzzy rules of the controller are represented in Table 1 and composed as follows:

IF 'E' **IS** 'A_{1i}' **AND** 'ΔE' **IS** 'A_{2i}' **THEN** 'U_{s1}' **IS** 'Z_i'
Where A_{1i}, A_{2i}, and Z_i ∈ {NB, NM, NS, ZE, PS, PM, PB } and i = 1, 2, ..., 7.

Table 1. *The rule base for the fuzzy steering controller*

E↓ΔE→	NB	NM	NS	ZE	PS	PM	PB
NB	NB	NB	NB	NB	NM	NS	ZE
NM	NB	NB	NB	NM	NS	ZE	PS
NS	NB	NM	NS	NS	ZE	PS	PS
ZE	NB	NM	NS	ZE	PS	PM	PB
PS	NM	NS	ZE	PS	PS	PM	PB
PM	NS	ZE	PS	PM	PB	PB	PB
PB	ZE	PS	PM	PB	PB	PB	PB

The inference mechanism uses minimum-maximum operation. The defuzzification is performed by the center of area method, using following formula:

$$CoA = \frac{\int_{x_{min}}^{x_{max}} f(x)\,x\,dx}{\int_{x_{min}}^{x_{max}} f(x)\,dx} \quad (8)$$

Where CoA is the center of area, x is the value of the output linguistic variable, and x_{max} and x_{min} represent the range of the linguistic variable [30].

4.2 The self-tuning fuzzy PID controller

As described in fig. 7, the fuzzy self-tuning PID controller consists of an adjustable parameter PID controller and a fuzzy tuning mechanism. The PID controller, which generates a control $U_{s2}(t)$ based on the parameter E(t), has the standard form as follows:

$$U_{s2}(t) = K_p E(t) + K_i \int E(t)\,dt + K_d \frac{dE(t)}{dt}$$
$$K_p = K_{p0} + \Delta K_p \quad (9)$$
$$K_i = K_{i0} + \Delta K_i$$
$$K_d = K_{d0} + \Delta K_d$$

Where K_p, K_i, K_d are, respectively, the proportional gain, the integral gain, and the derivative gain of the controller. The parameters K_{p0}, K_{i0}, K_{d0} (10, 0.02, 2.5) are the initial estimates of K_p, K_i, K_d, respectively, which are calculated by the relay feedback method described in [31-34], and ΔK_p, ΔK_i, ΔK_d are adjustments of PID gains, respectively.

There are two common approaches, for the fine-tuning of the PID gains as represented in [35 - 37]. While the authors in [35, 36] proposed the modifying method of the PID gains with a single parameter, the others in [37] used expert systems, which control ΔK_p, ΔK_i, ΔK_d independently. In this work, the second approach is used because that has more degree of freedom in controlling parameters. The three parameters, ΔK_p, ΔK_i, ΔK_d, are controlled by three independent fuzzy controllers, which have the same structure and inputs as the steering controller as mentioned in the previous section. The output variables of these controllers are ΔK_p, ΔK_i, ΔK_d, which also have a set of seven linguistic terms {NB, NM, NS, ZE, PS, PM, PB} as indicating in figure 11. In which, the values of A_{1-7} are {-3; -1.5; -0.5; 0; 0.5; 1.5; 3}, {-0.02; -0.015; -0.005; 0; 0.005; 0.015; 0.02}, {-2; -1.5; -0.5; 0; 0.5; 1.5; 2} corresponding to the membership function setting of the linguistic variables, ΔK_p, ΔK_i, ΔK_d, respectively. These values determined by using the trial and error method and the fact that the range of ΔK_d should be smaller than ΔK_p in order to avoid the oscillations; on the other hand, ΔK_i using to correct the steady state error plays a minor role when the process is reaching the set point.

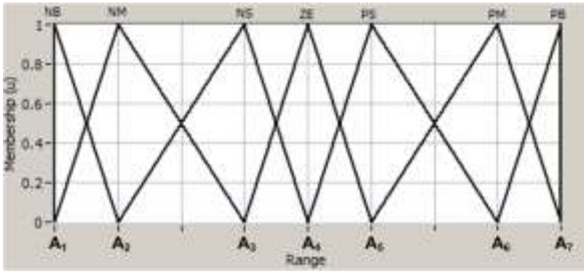

Figure 11 *The membership functions of (ΔK_p, ΔK_i, ΔK_d) variables*

The fuzzy rules are made by analyzing the influence of each parameter on the system output, which is summarized in Table 2 [38], experiments and corrections.

Table 2. *The effects of independent K_p, K_i, K_d tuning*

System Response	Rise Time	Overshoot	Settling Time	Steady State Error
K_p ↑	↓	↑	Small ↑	↓
K_i ↑	Small ↓	↑	↑	Large ↓
K_d ↑	Small ↓	↓	↓	Minor ↕

Where ↑, ↓, ↕ denote increase, decrease, and change, respectively.

In addition, when considering a typical step response regarding the sign and the area of (E, ΔE) as represented in figure 12, the fundamental principles of adjusting PID parameters describes as follows:

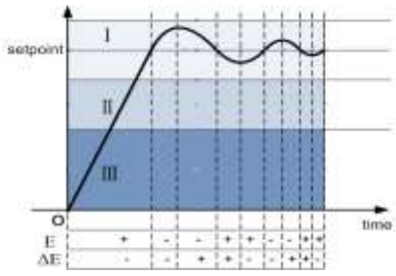

Figure 12 *A typical step response*

If |E| is relatively large (III), for accelerating the response rate of the system, regardless of the stage of the robot movement, ΔK_p should be rapidly increased, and ΔK_d should be slowly increased. Otherwise, if |E| and |ΔE| are of medium size (II), in order to reduce overshoot and absolute error of the system, ΔK_p should be smaller, ΔK_d have to be applied with moderate values, and ΔK_i should be appropriate. When the robot moves to the position at which |E| is close to the desired value (I), the small ΔK_p, ΔK_i should be provided so as to the system have good steady state performance. Simultaneously, to avoid the system's oscillation and reduce overshoot, when E and ΔE are both positive or negative, ΔK_d can be large. On the other hands, when E and ΔE are positive and negative, respectively, ΔK_d should be small. It is the same for the case in which E and ΔE are negative and positive.

After many trials with the reasoning method mentioned above, the fuzzy rules for three parameters ΔK_p, ΔK_i, ΔK_d are performed on rules table as shown in Table 3 to Table 5.

Table 3. *The rule base for* ΔK_p

E↓ΔE→	NB	NM	NS	ZE	PS	PM	PB
NB	PB	PB	PB	PM	PS	PS	ZE
NM	PB	PB	PM	PM	PS	ZE	ZE
NS	PM	PM	PM	PS	ZE	NS	NM
ZE	PM	PS	PS	ZE	NS	NM	NM
PS	PS	PS	ZE	NS	NS	NM	NM
PM	ZE	ZE	NS	NM	NM	NM	NB
PB	ZE	NS	NS	NM	NM	NB	NB

Table 4. *The rule base for* ΔK_i

E↓ΔE→	NB	NM	NS	ZE	PS	PM	PB
NB	NB	NB	NB	NM	NM	ZE	ZE
NM	NB	NB	NM	NM	NS	ZE	ZE
NS	NM	NM	NS	NS	ZE	PS	PS
ZE	NM	NS	NS	ZE	PS	PS	PM
PS	NS	NS	ZE	PS	PS	PM	PM
PM	ZE	ZE	PS	PM	PM	PB	PB
PB	ZE	ZE	PS	PM	PB	PB	PB

Table 5. *The rule base for* ΔK_d

E↓ΔE→	NB	NM	NS	ZE	PS	PM	PB
NB	PM	PM	ZE	ZE	ZE	PS	PS
NM	NS	NS	NS	NS	ZE	NS	PM
NS	NS	NB	NM	NS	ZE	ZE	PM
ZE	NB	NM	NM	NS	ZE	PS	PM
PS	NB	NM	ZE	NS	PM	PS	PS
PM	NM	NS	NS	NS	ZE	PS	PS
PB	PS	PS	ZE	ZE	ZE	PB	PB

4.3 The fuzzy velocity controller

As mentioned in section 4, this controller is used for safe tracking of a continuous path/line on the flat ground.

In [24], the authors indicated that the angular velocity, ω, of the differential-drive wheeled robot can be expressed in terms of the linear velocity, V, and the steering angle, θ, as follows

$$\omega = V.f(d,\theta) \tag{10}$$

In addition, with a kinematic limit of the mobile robot, the angular velocity, ω, also has a limited value such as ω_{max}. Therefore, to keep ω within the safe kinematic limits, one way is to control the linear velocity, V. In order to achieve the target, a fuzzy controller with the same inputs, structure, inference mechanism, and defuzzification method as the fuzzy steering controller, Section 4.1, is performed.

The output variable of the controller, U_v, has a set of four linguistic terms {ZE, PS, PM, PB} as shown Figure 13. The reason for this is that the steering controller is more important than the velocity controller, which has the main purpose of safe tracking. As a result, only four terms are used so as to reduce the complexity and the computational time of the system.

The rule base of the fuzzy velocity controller is given in Table 6, in which each one emulates human driving behavior such as "If distance error (E) is *large* and the trend is *moving far away the path/line*, then speed is *low*".

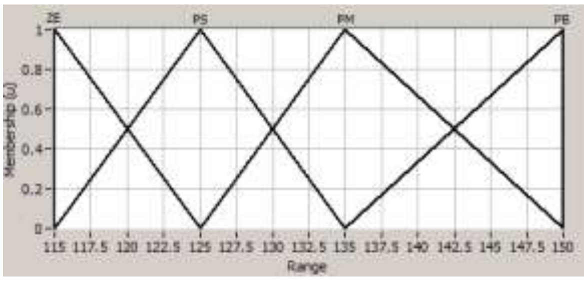

Figure 13 *The membership functions of* U_v *variable*

Table 6. *The rule base for* U_v

E↓ΔE→	NB	NM	NS	ZE	PS	PM	PB
NB	ZE	ZE	ZE	ZE	ZE	ZE	ZE
NM	ZE	ZE	PS	PS	PS	ZE	ZE
NS	ZE	PS	PS	PM	PS	PS	ZE
ZE	ZE	PS	PM	PB	PM	PS	ZE
PS	ZE	PS	PS	PM	PS	PS	ZE
PM	ZE	ZE	PS	PS	PS	ZE	ZE
PB	ZE	ZE	ZE	ZE	ZE	ZE	ZE

5. Experimental results

5.1 The mobile robot platform

The experimental mobile robot system is shown in Fig. 1. The physical parameters of the mobile robot are as follows. With reference to Fig. 2, the main drive wheels have the radius R = 0.05 m and mounted on an axel with the length of L = 0.35 m. The distance between the wheel´s axis and the central row of ROI, d, is 0.55 m. In all our experiments, the linear forward velocity, V, has the maximum value of 0.12 m/s.

5.2 Control system architecture of the robot

The system configuration of the experimental mobile robot that is used to deploy the path/line tracking algorithm is illustrated in Figure 14. The control algorithms are programmed on the LabVIEW environment and run with the sampling time of 80 ms. After the PC (Core i3, Windows 7) receives the images from the USB camera and does the algorithms, it passes the information about the duties of PWM signals to DAQ Card via RS232 communication. These PWM signals will be used to provide the voltage commands for two DC motors.

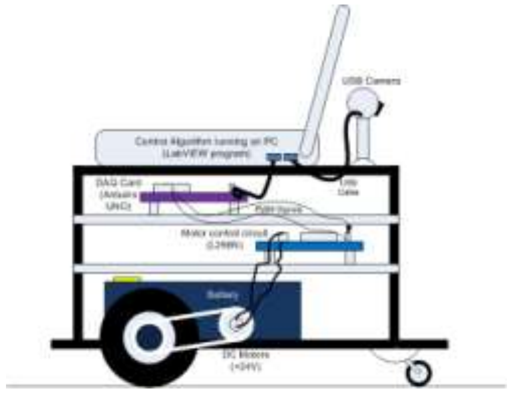

Figure 14 *The system configuration of the experimental mobile robot*

5.3 Test course

To validate the proposed control algorithm, a predefined path is constructed as represented in Figure 15. The path has a major turn and several small curves to check the output variation, which is recored during the moving process of the mobile robot.

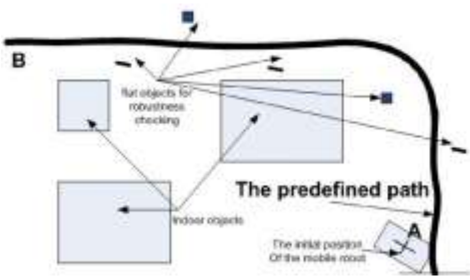

Figure 14 *Predefined paths of experimental results*

5.4 Experimental results

This section presents a comparison of the PID controller, fuzzy controller, and the hybrid fuzzy PID controller in tracking the predefined path, with the same conditional environment.

To measure the quality of each control algorithm, first the RMS value of the tracking error - the parameter E, (the average tracking performance), which is indicated by the formula (11), is used. In addition, the duration that the robot needs to travel from A to B, is also measured for the three controllers.

$$\|E\|_{rms} = \sqrt{\frac{1}{T}\int_0^T |E|^2\, dt} \quad (11)$$

The results are shown in Table 7 and Figure 15 to Figure 17.

Table 7. *Experimental results*

Controller	Time (s)	$\|E\|_{rms}$(m)
PID	47.6	0.0732
Fuzzy	34.8	0.0355
Hybrid Fuzzy PID	30.8	0.0277

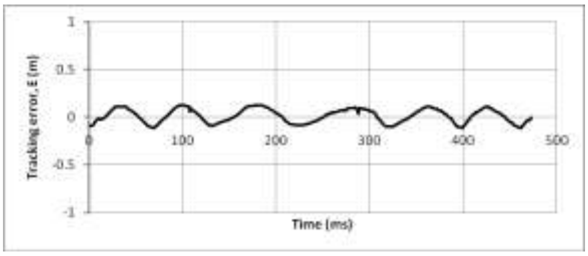

Figure 15 *Tracking error of PID controller*

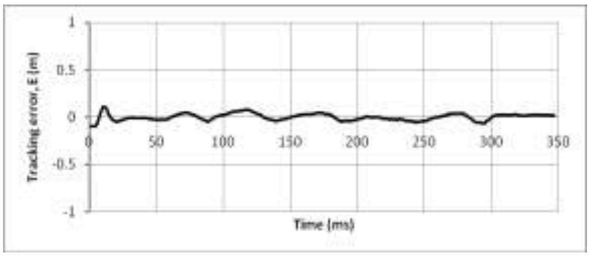

Figure 16 *Tracking error of Fuzzy controller*

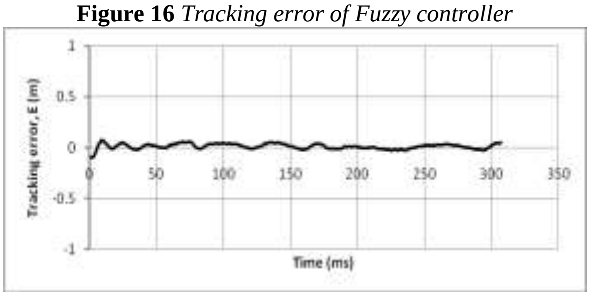

Figure 17 *Tracking error of Hybrid Fuzzy PID controller*

If the parameter E is positive, it means the mobile robot is on the left of the path. In contrast, if the tracking error, E, is negative, the robot is on the right of the path.

From the above results, it can be said that the proposed control scheme in follow-path/line problem is effective.

6. Conclusion

This work presents a combination of image processing techniques and a developing fuzzy control algorithm. The proposed system consists of two main parts and each one has its own function with the purpose of improving the performance of the system. While the vision system is used for providing the knowledge of the predefined path/line ahead of the mobile robot, the control algorithm produces an appropriate command so that the mobile robot can track the path successfully. The reported experimental results did prove the effectiveness of the proposed system. In future work, we propose to apply the control algorithm to another platform instead of the PC in order to improve the response time as well as the speed of the mobile robot.

References

[1] Gianluca Antonelli, Stefano Chiaverini, Giuseppe Fusco: *A Fuzzy logic based approach for mobile robot path tracking*, IEEE Transactions on Fuzzy systems, vol. 15, No.2, April 2007.

[2] Bachir Lakehal, Yacine Amirat, Jean Pontnau: *Fuzzy steering control of a mobile robot*, Industrial Automation and Control: Emerging Technologies, International IEEE/IAS Conference on, May 1995.

[3] L. Lapierre, D. Soetanto, A. Pascoal : *Adaptive vision-based path following control of a wheeled robot*, Proceeding of ECC2003 Conference, 2003.

[4] Jih-Gau Juang, Ko-Jui Hsu, and Chih-Min Lin: *A wheeled mobile robot path following system based on image processing and adaptive CMAC*, Journal of Marine Science and Technology, Vol. 22, No. 3, pp. 331-340, 2014.

[5] Jiang Zhong Ping, Henk Nijmeijer: *Tracking control of mobile robot: A case study in backstepping*, Automatic, Vol. 33, No. 7, pp.1393-1399, 1997.

[6] Wu Wei Guo, Chen Hui Tang, Wang Yue Juan: *Global trajectory tracking control of mobile robots*, Acta Automatica Sinica, Vol. 25, pp.877-888, 2001.

[7] Spandan Roy, Sambhunath Nandy, Ranjit Ray, and Sankar Nath Shome: *Time Delay Sliding Mode Control of Nonholonomic Wheeled Mobile Robot: Experimental Validation*, IEEE International Conference on Robotics & Automation, June 7, 2014.

[8] Tan Lain Chung, Trong Hieu Bui, Tan Tien Nguyen, Sang Bong Kim: *Sliding Mode Control of Two-Wheeled Welding Mobile Robot for Tracking Smooth Curved Welding Path*, KSME International Journal, VoL 18 No. 7, pp. 1094~1106, 2004.

[9] Chih-Lyang Hwang, Hsiu-Ming Wu: *Trajectory tracking of a mobile robot with frictions and uncertainties using hierarchical sliding-mode under-actuated control*, IET Control Theory and Applications, February 2013.

[10] Umar Farooq, K. M. Hasan, Athar Hanif, Muhammad Amar, and Muhammad Usman Asad: *Fuzzy logic based path tracking controller for wheeled mobile robots*, International Journal of Computer and Electrical Engineering, Vol. 6, No. 2, April 2014.

[11] Umar Farooq, Muhammad Amar, Muhammad Usman Asad, Ghulam Abbas, and Athar Hanif: *Fuzzy logic reasoning system for line following robot*, IACSIT International Journal of Engineering and Technology, Vol.6, No.4, August 2014.

[12] M. Nasir Ayob, M. S. M Hashim, M. N Ainuddin, Khairunizam Wan, Rudzuan M. Nor: *Design navigation error reduction algorithm for mobile robot based on fuzzy logic control*, Australian Journal of Basic and Applied Sciences, 8(4) Special 2014, Pages: 532-540.

[13] Bai Mingyou, Wang Haiying Xu Rui, Jia Jinlin: *Optimization design for PID parameters of mobile robot based on generic algorithm*, 2009 Sixth International Conference on Fuzzy Systems and Knowledge Discovery, 2009.

[14] Ramzy S. Ali, Ammar A. Aldair, Ali K. Almousawi: *Design an optimal PID controller using artificial bee colony and genetic algorithm for autonomous mobile robot*, International Journal of Computer Applications, Vol. 100, No. 16, August 2014.

[15] Yina Wang, Shuoyu Wang, Renpeng Tan, and Yinlai Jiang: *Adaptive control method for path tracking of wheeled mobile robot considering parameter changes*, International Journal of Advanced Mechatronic Systems, Vol. 4, No. 1, 2012.

[16] R. Hartley, A. Zisserman: *Multiple view geometry in computer vision*, Cambridge University Press, 2000;

[17] Hiroshi Kano, Keisuke Asari, Yohei Ishii, Hitoshi Hongo: *Precise top view image generation without global metric information*, IEICE Transaction of Information and Systems, Vol. E91-D, No. 7, July 2008.

[18] "*Searching and matching VIs*", chapter of "*NI Vision Concepts Help*", online resource at http://digital.ni.com/manuals.nsf/websearch/16712B9E3D2179698625776E0057700D.

[19] Ballard, D. H. & Brown, C. M.: *Computer Vision*, Prentice Hall, New Jersey, 1982.

[20] Jain, R., Kasturi, R. & Schunk, B.G.: *Machine Vision*, McGraw Hill, New York, 1995.

[21] H. A. Mallot, H. H. Biilthoff, J. J. Little, S. Bohrer: *Inverse perspective mapping simplifies optical flow computation and obstacle detection*, Biological Cybernetics, pp. 177-185, 1991.

[22] Anuar Mikdad Muad, Aini Hussain, Salina Abdul Samad, Mohd. Marzuki Mustaffa, Burhanuddin Yeop Majlis: *Implementation of inverse perspective mapping algorithm for the development of an automatic lane tracking system*, TENCON 2004. 2004 IEEE Region 10 Conference, Volume A, November 2004.

[23] Edwin Vans, Gancho Vachkov, Alok Sharma: *Vision based autonomous path tracking of a mobile robot using fuzzy logic*, Computer Science and Engineering (APWC on CSE),

2014 Asia-Pacific World Congress on, November 2014.

[24] Mahit Gunes, A Fevzi Baba: *Speed and position control of autonomous mobile robot on variable trajectory depending on its curvature*, Journal of Scientific & Industrial Research, Vol. 68, pp. 513-521, June 2009.

[25] Amitava Chatterjee, Anjan Rakshit, N. Nirmal Singh: *Vision Based Autonomous Robot Navigation: Algorithms and Implementations*, Chapter 6, Springer Verlag Berlin Heidelberg 2013.

[26] F. Cardoso, F. Fontes, C. Pais, P. Oliveira, M. I. Ribeiro: *Fuzzy logic steering controller for a guided vehicle*, Proceedings of the IEEE 7th Mediterranean Electrotechnical Conference, MELECON 94, page 711-714, Turkey, April 1994.

[27] Aditi Barua, Lalitha Snigdha Mudunuri, and Olga Kosheleva: *Why trapezoidal and triangular membership functions work so well: Towards a theoretical explanation*, Journal of Uncertain Systems, Vol. 8, 2014.

[28] Tzung-Pei Hong, Chai-Ying Lee: *Introduction of fuzzy rules and membership functions from training examples*, Fuzzy Sets and Systems 84(1996) 33-47, 1996 Elsevier Science B.V.

[29] M. Braae, D. A. Rutherford: *Selection of parameters for a fuzzy logic controller*, Fuzzy Sets Syst., vol. 2, no. 3, pp. 185-199, 1979.

[30] "*LabVIEW PID and Fuzzy Logic Toolkit User Manual*", online resource at http://www.ni.com/pdf/manuals/372192d.pdf

[31] C. C. Hang, K. J. Astrom: *Practical aspects of PID auto-tuners based on relay feedback*, Proc. Intern. Sysp. Adaptive Control of Chemical Processes, Denmark, pp. 149-154, 1988.

[32] K. J. Astrom, T. Hagglund: *An industrial adaptive PID controller*, Proc. 1989 IFAC Symp. Adaptive System in Control and Signal Processing, pp. 293-298, 1989.

[33] Shi-Zhong He, Shaohua Tan, Feng-Lan Xu, Pei-Zhuang Wang: *Fuzzy self-tuning of PID controllers*, Fuzzy Sets and Systems 56 (1993) 37-46, 1993- Elsevier Science Publishers B.V.

[34] Hyung-Soo Hwang, Jeong-Nae Choi, Won-Hyok Lee, Jin-Kwon Kim: *A tuning algorithm for the PID controller utilizing fuzzy theory*, Neural Networks, 1999. IJCNN'99. International Joint Conference on, Vol. 4, July 1999.

[35] Jianming Zhang, Ning Wang, Shuqing Wang: *A developed method of tuning PID controllers with fuzzy rules for integrating processes*, Proceeding of the 2004 American Control Conference, Boston, Massachusetts, July 2, 2004.

[36] Wang Xiao-Kan, Sun Zhong-Liang, Wanglei, Feng Dong-qing: *Design and research based on fuzzy PID parameters self-tuning controller with Matlab*, 2008 International Conference on Advanced Computer Theory and Engineering.

[37] Zhongbao Luo, Wei Li: *Tracking of mobile robot expert PID controller design and simulation*, 2014 International Symposium on Computer, Consumer and Control.

[38] Kiam Heong Ang, Gregory Chong, Yun Li: *PID control system analysis, design, and technology*, IEEE Transactions on Control Systems Technology, Vol. 13, No. 4, July 2005.

Pham Xuan Thuy was born in 1982. He received a degree of Master in Intelligent Electronics of GroupT International University College Leuven in 2013. From 2013 to present, he conducts as an assisstant lecturer in Le Quy Don Technical University, in Automatic Control and Computing Department in the Automatic Control Engineering Falcuty. His research interests include robotics, control systems, digital design, image processing and embedded systems.

Nguyen Tang Cuong was born in 1950, Ass. Professor, Ph.D, Lecture in Le Quy Don Technical University. He received University Education, Ph.D Degree and Post-graduate Technology Practics from Minsk Air Defence Missile Engineering University, and Bauman Moscow State Technical University. He published more than 80 scientific papers in technical related filds of Automation, Flight Equipment Control, Computer System, Signal Processing and Industrial Control System.