A Method for Path Planning Strategy and Navigation of Service Robot

Widodo Budiharto[2*], Ari Santoso[1†],
Djoko Purwanto[1‡] Achmad Jazidie[1§]

1   Institute of Technology Sepuluh Nopember,
Surabaya, Indonesia

2   BINUS University, Jakarta, Indonesia

Abstract

This paper presents our work on the development of Path Planning Strategy and Navigation by using ANFIS(Adaptive Neuro-Fuzzy Inference System)controller for a vision-based service robot. The robot will deliver a cup to a recognized customer and a black line as the guiding track for navigating a robot with a single camera. The contribution of this research includes a proposed architecture of ANFIS controller for vision-based service robot integrated with improved face recognition system using PCA, and the algorithm for moving obstacle avoidance. We also propose a path planning algorithm based on Dijkstra's algorithm to obtain the shortest path for robot to move from the starting point to the destination. In order to avoid moving obstacles, we have proposed an algorithm using binaural ultrasonic sensors. The service robot called Srikandi is also equipped with 4 DOF arm and a framework of face recognition system. The proposed path planning strategy and navigation was tested empirically and proved effective.

Keywords

service robot · navigation · ANFIS · path planning · face recognition

1. Introduction

Vison-based service robots have recently made rapid progress to tackle more complex manipulation activities such as, for example, delivering a cup to an identified customer in restaurant environment.

Path planning strategy and navigation is an essential issue in robot and artificial intelligence. Path planning for mobile robot in dynamic environments has been studied extensively in [1][2], however, vision-based path planning for service robot has not been fully explored. Vision sensor for navigation has good sensing ability and makes robots able to identify and recognize the objects near them. Navigation for mobile robot can be categorized as indoor navigation and outdoor navigation. Indoor navigation can be done with a map or without a map [3].

Navigation using a line for path tracking is one of the solution for dedicated service robot. In doing so, the robot could be implemented to any restaurants that have different landmark and environment [4]. However, the previous research [4] uses only a single track and there is no ablity to avoid the obstacles. In this paper, we have made improvement on [4] with proposed Path planning strategy, obstacles avoidance and improved ANFIS controller for indoor navigation without global map which is based on line tracking. Path planning will be used by a service robot to find the shortest path from home to the goal.

Dijkstra algorithm can be used for path planning [5][6][7]. In many research, fuzzy logic and Neural networks can be used for robot's navigation. Fuzzy logic can be used as a controller for robot's navigation in an unknown but dynamic environment. Li and Yang[8] proposed an obstacle avoidance approach using fuzzy logic, with the input sensors separately inferred. Neural networks are model-free systems that are organized in a way to simulate the cells of a human brain. Neural networks have a self-learning and self-tuning capability, and therefore can be used to model various systems. Therefore, fuzzy logic and neural networks can be combined to solve the complex robot navigation control problem and therefore improve the performance [10].

Certain fuzzy systems are universal function approximators. In order to identify a suitable fuzzy system for a given problem, membership functions (parameters) and a rule base (structure) must be specified. This can be done by prior knowledge, by learning, or by combination of both. If a learning algorithm that uses local information is applied, it will causes local modifications in a fuzzy system, and this approach is usually called neuro-fuzzy system [9]. Anmin Zhu et al. [10] presented a neuro-fuzzy controller for robot navigation, where the robot uses GPS and many distance sensors (9 sensors) to obtain the distance from the robot to the obstacle. In this paper, we solved the navigation problem and the problem of sensor limitations using camera presented in section 2, 3 and 4. Section 5 and 6 explain the proposed ANFIS architecture and Face recognition system. Finally, results and discussion from the implementation of the proposed method and algorithms on the service robot Srikandi I, are presented in section 7.

2. Architecture of vision-based service robot

We have proposed the model of service robot as shown in fig. 1. Because this robot need to deliver a cup using path tracking, to recognizes and tracks people, many supporting functions must be developed and integrated such as face recognition system, path tracking and mov-

*E-mail: widodo@widodo.com
†E-mail: santoso@ee.its.ac.id
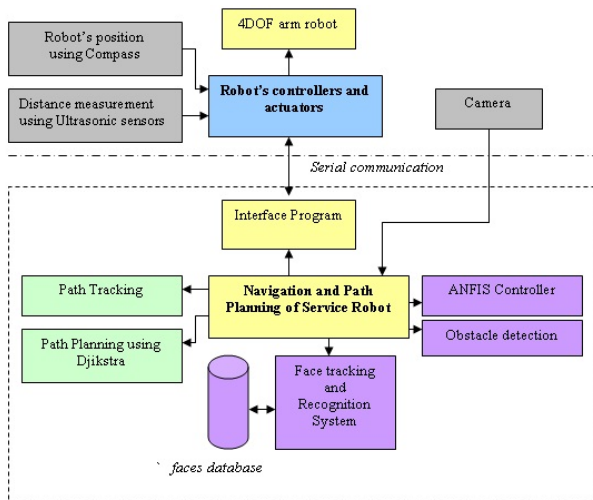‡E-mail: djoko@ee.its.ac.id
§E-mail: jazidie@elect-eng.its.ac.id

Figure 1. *An overview of Vision-based service robot called Srikandi I.*

Figure 2. *The visual-perception model for vision-based service robot. After interpretation process, then the information is used for navigating the robot or deciding other actions for the robot to take, such as directs a robot to customer's position.*
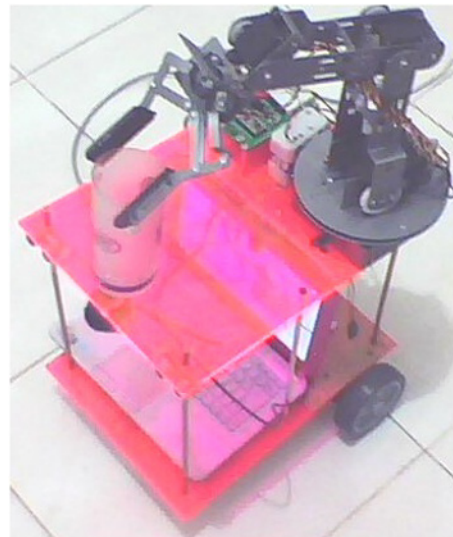

Figure 3. *Vision-Service robot called Srikandi I.*

ing obstacles detection. We have developed efficient Faces database used for face recognition system in order to recognize customer based on PCA. For robot navigation, Face, line, cup and obstacles are used to be processed by the ANFIS controller, then the output is the speed of the left and right motor and action command for the arm robot. There is interface program between Laptop for coordinating robot controller and text to speech controller are more interactive with user.

Robot will detect face and checking whether the face is customer or not using simple faces database program. An ultrasonic sensor used to detect whether a cup has been already loaded to the robot. When the cup is loaded to the robot, the robot starts to go by following the line until it reaches the end of the line and the camera has caught the person's face. When face is detected, arm robot will give the cup to that person with predefined motion, and then robot will return to the home. Binaural ultrasonic sensors [21] is used for detect moving obstacles. If sensors detect obstacles, robot will stop for a moment, then check again if obstacles still exist or no. If obstacles come nearer to robot, the robot will move backward, otherwise the robot will continue moving.

Among the indoor service robots that use camera, and those are able to operate in the environments with humans, and especially those who are able to interact with the customer, have gained high interest in recent years. Visual perception is the ability to interpret the information and surroundings from the effects of visible light reaching the eye. The resulting perception is also known as eyesight, sight, or vision. Visual or perception-based service robot for customer identification, is an interpretation process to directs a service robot to a destination of identified customer, based on face recognition system. After interpretation of images, the information is then used by the robot to decide on what actions to take, based on the task given by a developer. The basic visual--perception model for a service robot is shown in fig. 2.

Fig. 3 presents our robot named Srikandi used in this experiment, where single camera is used as line sensor to identify customer, and ultrasonics sensors are for obstacles detection.

3. Path planning strategy

Path planning problems in various environments have been studied extensively in the last two decades. Some exact algorithms are proposed, but with exponentially increasing complexity [12]. The goal of path planning for robot in a space with obstacles, is to find a collision-free path of robot from the starting to the target position. There are many approaches based on types of obstacles, dimensionality of the space and restrictions for robot movements. Among the most frequently used ones are roadmap methods (visibility graphs, Voronoi diagrams, rapidly exploring random trees) and the method based on cell decomposition[13][15][20].

Uniform cost algorithm proposed by Dijkstra is a sequential method that can efficiently compute the optimum path from a reference node to any arbitrary node for a given cost map. For a given source vertex (node) in the graph, the algorithm finds the path with lowest cost (i.e. the shortest path) between that vertex and every other vertex [14].

Let $G = (V, E)$ be a weighted digraph, with weight function $\omega : E->$R mapping edges to real-valued weights. If $e = (u, v)$, we write $\omega(u, v)$ for $\omega(e)$. The length of a path $p = \langle v_0, v_1..., v_k \rangle$ is the sum of the weights of its consituent edges:

$$length(p) = \sum_{i=1}^{k} \omega(v_{i-1}, v_i) \tag{1}$$

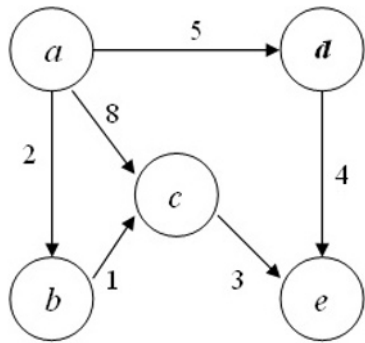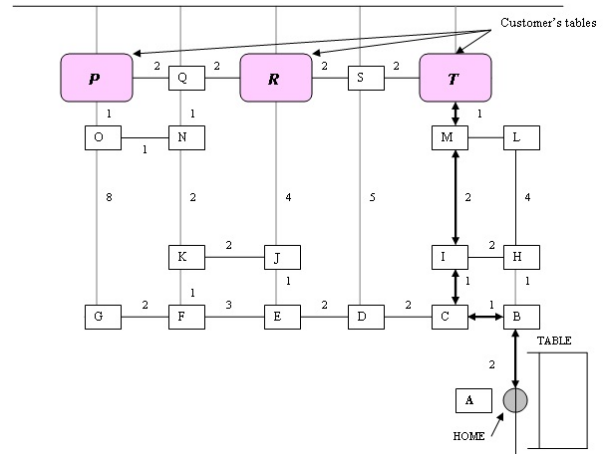Figure 4. Example of shortest path using Dijkstra`s algorithm, Distance (<a,b,c,e>) is 6.

(a)

(b)

Figure 5. Testing map for path planning, for the shortest path, robot will move from A, B, C, I and M, and return with the reverse order (a). Each of customer`s table identified by robot by the θ produced between robot position and table P, R or T (b).

The distance from u to v, denoted $\delta(u, v)$, is the length of the minimum length path if there is a path from u to v; and is ∞ otherwise. For example, the minimum path in fig. 4 from a to e (a,b,c,e) is 6.

Because of the ability of Dijkstra`s algorithm to compute the shortest path, we applied it to compute the shortest path from the starting point of robot to the customer table. Fig. 5(a) shows the testing map for robot. P, R and T is customer table. In fig. 5(b), each of customer`s table identified by robot by the θ produced between robot position and table P, R or T using compass sensor. For example, if customer order a cup of coffee and then sitted on the table T, then to robot from HOME should move from A to B, C, I and M as the shortest path, and go back to the home using reverse order .
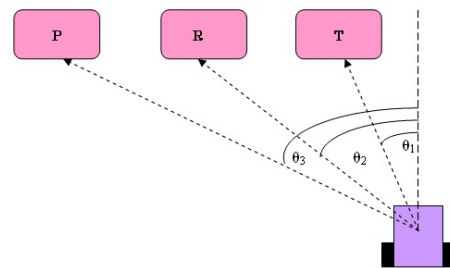
The proposed algorithm for path planning strategy using Dijkstra algorithm for this case in fig. 5.

Algorithm 1. Path planning strategy

1. Detect face

2. say welcome to the customer

3. Check whether a cup is loaded

4. if cup loaded and face detected

5. go to node B

6. // Find the position using compass

7. scanning customer`s table

8. find the θ between robot and table

9. // using Dijkstra`s algorithm

10. calculate shortest path

11. while (not last node) do

12. move to next node

13. endif

14. giving the cup

15. Back to home

When the robot is running, the robot also checks the obstacles using 2 ultrasonics sensors installed on the left and right side of the robot. In this experiment, the delay 1 second enough for robot to stop and the minimum distance 20cm is enough to avoid collision with the obstacle. The algorithm for detecting moving obstacles is shown below :

Algorithm 2. Moving obstacles detection

1. initialize sensors

2. while (running) do

3. Read sensor right & left

4. if (right ßßleft) <=20cm then

5. Stop

6. Wait 1 second

7. Read sensors

8. if (rightßßleft) <15cm then

9. Stop

10. Backward

11. else

12. forward

13. endif

14. endif

Figure 6. *ANFIS architecture using 2 input x and y[15].*
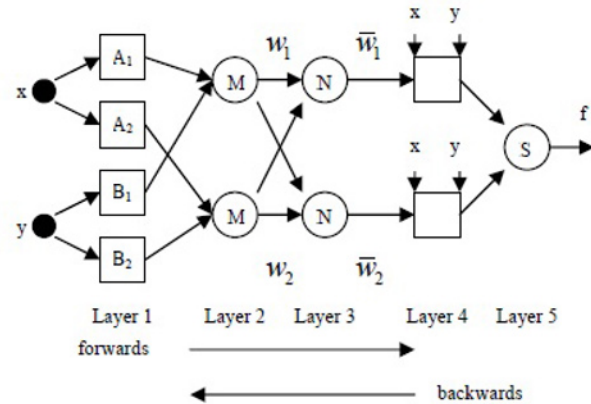
4. Adaptive neuro fuzzy inference system (ANFIS)

Neuro-Fuzzy techniques have emerged from the fusion of neural network (ANN) and Fuzzy Inference System (FIS) and form a popular framework for solving real word problem[16]. Fuzzy Logic Controllers (FLC) has played an important role in the design and enhancement of a vast number of applications. ANFIS are fuzzy Sugeno models put in the framework of adaptive systems to facilitate learning and adaptation. Such framework makes FLC more systematic and less relying on expert knowledge. ANFIS is a fuzzy inference system formulated as a feed-forward neural network. Hence, the advantages of fuzzy system can be combined with a learning algorithm. Let us consider two-fuzzy rules based on a first order Sugeno model to present the ANFIS architecture:

Rule 1 :

If x is A_1 and y is B_1 then

$$z_1 = p_1 * x + q_1 * y + r_1 \tag{2}$$

If x is A_2 and y is B_2 then

$$z_2 = p_2 * x + q_2 * y + r_2 \tag{3}$$

ANFIS architecture to implement these two rules is shown in Fig. 6. Note that a circle indicates a fixed node whereas a square indicates an adaptive node (the parameters are changed during training).

Layer 1: All the nodes in this layer are adaptive nodes, i is the degree of the membership of the input to the fuzzy membership function (MF) represented by the node :

$$O_{1,i} = \mu Ai\ (x) \qquad i = 1, 2 \tag{4}$$

$$O_{1,i} = \mu Bi\text{--}2\ (y) \qquad i = 3, 4 \tag{5}$$

A_i and B_i can be any appropriate fuzzy sets in parameter form. For example, if bell MF is used then :

$$\mu_{Ai}\ (x) = \frac{1}{1 + \left| \frac{x - c_i}{a_i} \right|^{2b_i}} \qquad i = 1, 2 \tag{6}$$

Where a_i, b_i and c_i are the parameters for the MF.

Layer 2: The nodes in this layer are fixed (not adaptive). These re labeled M to indicate that they play the role of a simple multiplier. The outputs of these nodes ar e given by:

$$O_{2,i} = w_i = \mu Ai\ (x)\ \mu Bi\ (y) \qquad i = 1, 2 \tag{7}$$

The output of each node in this layer represents the firing strength of the rule.

Layer 3: Nodes in this layer are also fixed nodes. These are labeled N to indicate that these perform a normalization of the firing strength from previous layer. The output of each node in this layer is given by :

$$O_{3,i} = \overline{w_i} = \frac{w_i}{w_1 + w_2} i = 1, 2 \tag{8}$$

Layer 4 : All the nodes in this layer are adaptive nodes. The output of each node is simply the product of the normalized firing strength and a first order polynomial :

$$O_{4,i} = \overline{w_i} \cdot f_i = \overline{w_i}\ (p_i x_1 + q_i y_1 + r_i) \tag{9}$$

Where pi, qi and ri are design parameters (consequent parameter since they deal with the then-part of the fuzzy rule).

Layer 5: This layer has only one node labelled S to performs a simple summer. The output of this single node is given by :

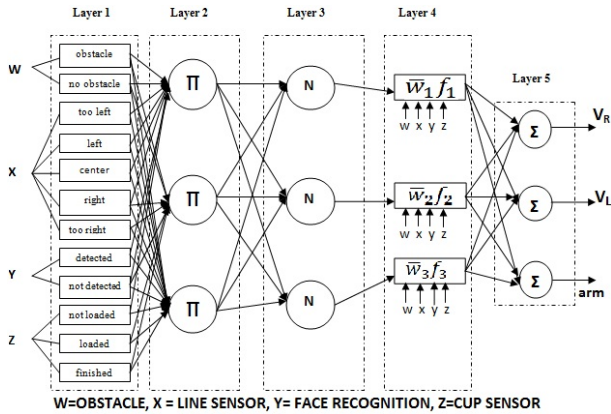$$O_5 = \sum \overline{w_i f_i} = \frac{\sum_i w_i f_i}{\sum_i w_i} \tag{10}$$

Figure 7. *Proposed ANFIS controller for Navigation of service robot.*

The ANFIS architecture is not unique. Some layers can be combined and still produce the same output. In this ANFIS architecture, there are two adaptive layers. Layer 1 has three modifiable parameters (ai, bi and ci) pertaining to the input MFs [16][17]. These parameters are called premise parameters. Layer 4 has also three modifiable parameters (pi, qi and ri) pertaining to the first order polynomial. These parameters are called consequent parameters. The task of training algorithm for this architecture is tuning all the modifiable parameters to make the ANFIS output match the training data [18]. Based on the hybrid learning algorithm proposed by Jang, Sun and Mizutani [17] which uses a combination of Steepest Descent and Least Squares Estimation (LSE), if the premise parameters are fixed the output is linear in the consequent parameters. So, the hybrid learning algorithm uses a combination of steepest descent and least squares to adapt the parameters in the adaptive network.

5. ANFIS controller for navigation

Fig. 7 is proposed ANFIS architecture for service robot. We give training data for ANFIS controller using mapping method. We designed 4 input for service robot; w for obstacle avoidance, x for robot position, y for face recognition and z for cup status. The training data contain 48 possible input (x,y,z) and appropriate output (velocity of right motor V_R, velocity of left motor V_L, action of 4DOF Arm robot). We only create 16 rules where each rule consist of 3 training data. Morphological operation (erosion followed by dilation) is applied to original image to eliminate noises. Then, we determine threshold value for black color as path on the floor. Final image includes the information about the center position, and detected position issued as an input test data for controller. Fig. 7 is an architecture of ANFIS controller with 4 input and 3 output for controlling speed of motors and arm robot.

We created rules for controller shown at Table 1. These rules determined how robot behaves. Target seeking is an action taken by the robot to keep running until it found the end line.

Table 1. *The rule base of the proposed fuzzy controller. C: center; L: left; TL: too left; R: right; TR: too right; S: stop; ND: face not detected;D: face detected; L: loaded; NL: not loaded; F: finished; PS: positive small; PB: positive big; NA: no action; A: action, NO: NO obstacles, X: don't care.*

Rule	Input				Output			Behaviour
	w	x	y	z	VL	VR	Arm	
1	NO	C	ND	NL	0	0	NA	stop
2	NO	C	ND	L	PS	PS	NA	Target seeking
3	NO	L	ND	L	PS	0	NA	Target seeking
4	NO	TL	ND	L	PB	0	NA	Target seeking
5	NO	R	ND	L	0	PS	NA	Target seeking
6	NO	TR	ND	L	0	PB	NA	Target seeking
7	NO	S	ND	L	0	0	NA	Target seeking
8	NO	S	D	L	0	0	A	Giving order
9	NO	S	D	NL	PB	0	NA	Return back
10	NO	C	ND	F	PS	PS	NA	Going home
11	NO	L	ND	F	PS	0	NA	Going home
12	NO	TL	ND	F	PB	0	NA	Going home
13	NO	R	ND	F	0	PS	NA	Going home
14	NO	TR	ND	F	0	PB	NA	Going home
15	NO	S	ND	F	0	0	NA	Available
16	O	x	x	x	x	x	x	Stop

Figure 8. *Original image (a), then applied preprocessing image to greyscale image (b) and histogram equalization (c).*

6. A framework for face recognition

We have developed a framework of Face recognition system for vision-based service robot and face database named ITS Face database. This framework is very usefull as a information for robot to identify a customer and what items are ordered by a customer. The single camera used in this research is 640x480 pixels. The size of face image is cropped to 92x112pixels using Region of Interest method (ROI) as shown in figure below. These images also used as training images for face recognition system. We use histogram equalization for contrast adjustment using the image's histogram. This method usually increases the global contrast of many images, especially when the usable data of the image is represented by close contrast values. Histogram equalization accomplishes this by effectively spreading out the most frequent intensity values.

For illumination variation modelling, we construct images under different illumination conditions by generate a random value for brightness level using this formula :
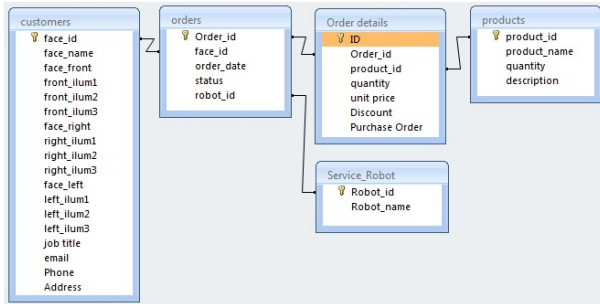
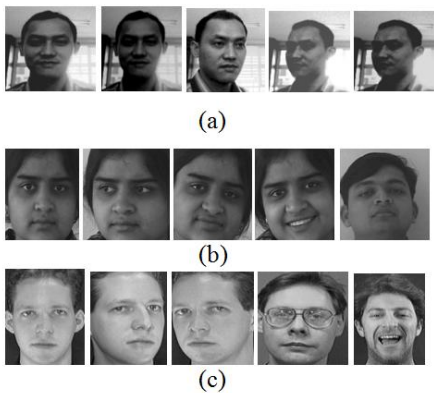Figure 9. *A proposed database for a Framework of face recognition system.*

(a)

(b)

(c)

Figure 10. *ITS, Indian and ATT face database used as comparation to see the effect of illumination at face recognition.*

$$I_o(x, y) = I_i(x, y) + c \qquad (11)$$

Where I_o is intensity value after brightness operation applied, I_i is intensity value before brightness operation and c is brightness level.

First, to store training faces of customers and items/products ordered, we have proposed a database that consists of a table **customers, products, orders, Service_robot** and **order detail** :

Each of face databases used in this experiment consists of 10 sets of people's face show in fig. 10. ITS face database consists of 3 poses (front, left, right) and varied with illumination. ATT face database consists of 9 differential facial expression and small occlusion (by glass) without variation of illumination. The Indian face database consists of eleven pose orientation without variation of illumination and the size of each image is too small than ITS and ATT face database.

The training process involved input images as training images and the result then stored in .xml file, which contain information of the name of customers, items ordered, and faces description will be used for testing phase. The training images stored in database can be view as shown in figure below :
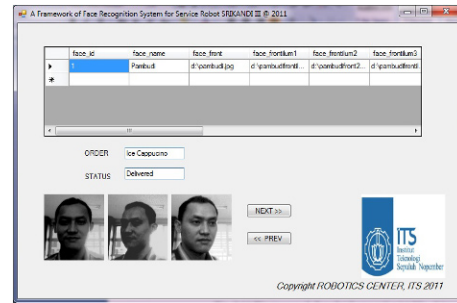
(a)
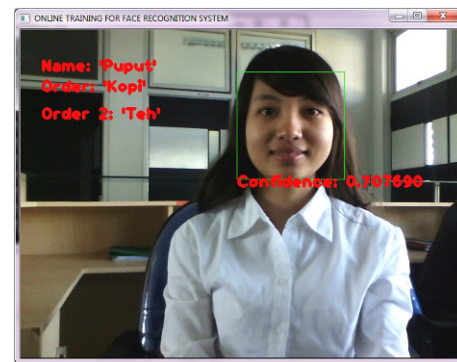
(b)

Figure 11. *An application interface for ITS face databases using 3 poses of images with illumination variation used for each person (frontal, left and right poses)(a) and result of our framework of face recognition displayed the name of the customer and her order Coffee and Tea (b).*

7. Result and discussions

Path planning strategy based on Dijkstra`s algorithm and ANFIS controller has been tested for navigating service robot to delivers a cup in a lab as simulation of indoor environment. Obstacles detection is done with 2 ultrasonic sensors. They determine the distance to the obstacle by measuring the time of fly from ultrasonic pulse. Periodically, information received by the sensors is used to detect obstacles. Face recognition based on eigenspaces can be used and databases for the images are developed. Fig. 10 shows a setup of experiment for this research.

Fig. 13 is a result for customer identification to direct a robot to the position of customer's table. Program succesfully identified a customer and his order using our proposed face recognition system.

We also evaluate the result of our proposed face recognition system and compared with ATT and Indian face database using Face Recognition Evaluator developed by Matlab. The success rate or accuracy comparison between 3 face database shown below:

Line sensor succesfully detects track and controller produced output with expected response. Robot position gained by reading line track on the landmark that separate into 5 region as shown in fig. 15. This region is used to adjust motor's speed. Each region has pixel range to

(a)
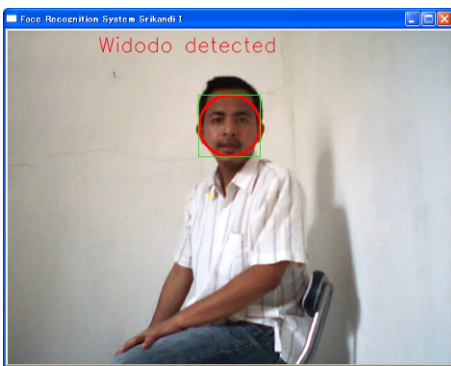
(b)

Figure 12. *A setup of experiment using tables, chairs and line tracker as a simulation of a cafe / fast food restaurant (a). Face recognized and detected using Face Recognition System developed using OpenCV (b).*

Figure 13. *An example of face detected and recognized succesfully using Haar Cascade Classifier and the information of the ordered product.*
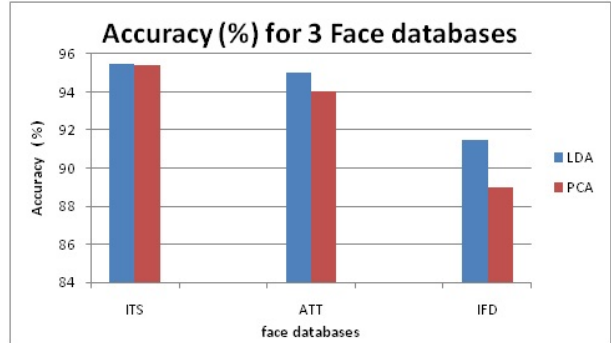

Figure 14. *Success rate comparison of face recognition between 3 face databases, each using 10 sets face. It shown clearly that ITS database have highest success rate than ATT and Indian face database (IFD)[22] when the illumination of testing images is varied. The success rate using PCA in our proposed method and ITS face database is 95.5 %, higher than ATT face database 95.4%.*
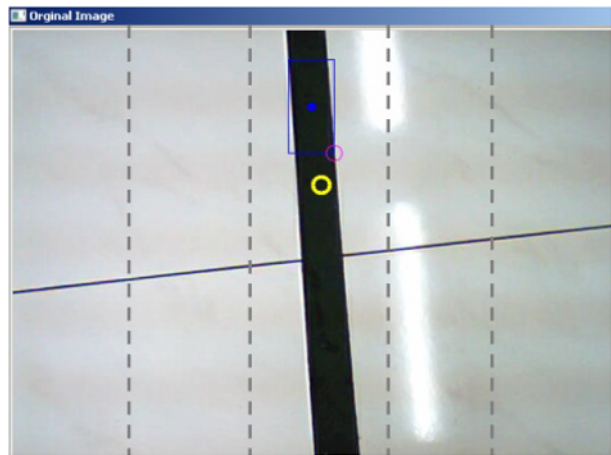

Figure 15. *Original image and points for input data, yellow circle is center position. Dashed lines separate 5 areas for controlling the speed of DC motors.*

indicate robot's position. After camera has detected line and obtaining robot's position, the images was used as an input to ANFIS controller to be processed. Fig. 15 also displays the track and the center position (indicated by yellow circle) as reference, and the robot's position for input data to controller.

Original images are then procesed using morphological operation (dilation, erosion and smoothing) for reducing the noises and then we got the region of interest as shown at Figure 16 using OpenCV.

Robot automatically runs when a cup is loaded to the robot, after that camera detects the line to give information to the robot about its position. We have developed program for ANFIS controller using C++ based on Adaptive Neuro Approximation method[19]. The testing data and prediction shown below indicated that the controller able to predict correctly.

Based on results that displayed using GnuPlot, after training process, we got the best error after epoch 62 as shown in Fig. 18.
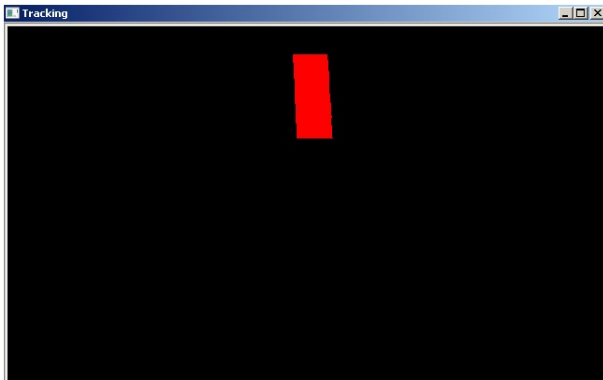
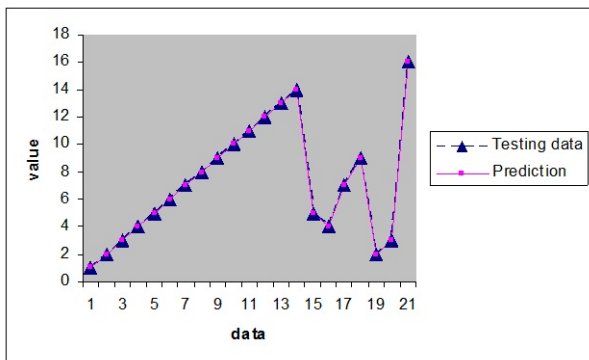**Figure 16.** *Region of interest from the black line.*

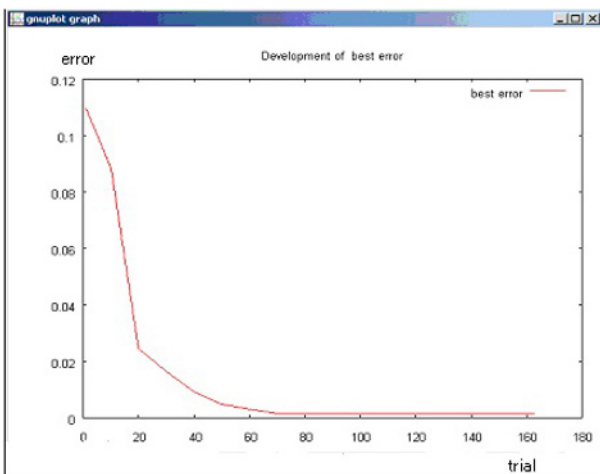**Figure 17.** *Testing data and prediction result using ANFIS.*

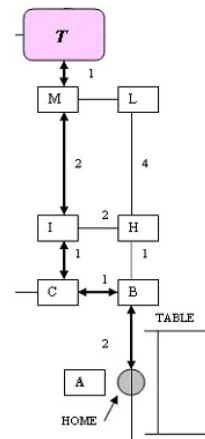**Figure 18.** *Development of best error starting at epoch 62.*

Figure 19. *An experimental result produced by the robot, robot successfully able to find the shortest path from point A to table T. Gray track is a path from home to the goal, and dashed line is a path when robot going back to home.*

Fig. 19 is an experimental result produced by the robot. It shown that robot accomplished the task using shortest path (from A, B, C, I, and M compared if robot have to move from A, B, H, L and M), so it gave an efficiency in time from HOME to goal.

8. Conclusions

This paper introduces a new method for Path planning and navigation of vision-based service robot. Path planning strategy using Dijkstra algorithm can be used for obtaining shortest path. ANFIS controller for navigation of robot is proposed and is implemented. ANFIS controller can be used as controller which only uses 16 rules and epoch requried for training the controller very small (62 epochs), so this system is very powerfull for service robot. camera can be used for face recognition and line sensor in service robot. For future work, we will implement a vision-based navigation and obstacles avoidance in dynamic environment using stereo vision.

References

[1] T. Tsubouchi and M. Rude, Motion planning for mobile robots in a time-variying environment : A Survey, Journal of Robotics and Mechatronics, vol. 8 no.1(1996), 15-24.

[2] Y.Z. Chang and R. P. Huang, A simple Fuzzy motion planning strategy for autonomous mobile robots, 33rd Annual Conference of the IEEE Industrial Electronics Society, Taiwan, 2008, 477- 480.

[3] D. Ito (ed), Robot Vision, Nova Science Publisher, 2009.

[4] W. Budiharto, D. Purwanto and A. Jazidie, Indoor Navigation using ANFIS controller for Servant Robot, 2nd IEEE International Conference on Computer Engineering and Its Application (ICCE 2010), Bali, 2010, 582.586. DOI: 10.1109/ICCEA.2010.119.

[5] H. Kang, B. Lee and K. Kim, Path Planning Algorithm Using the Particle Swarm Optimization and the Improved Dijkstra Algorithm, Workshop on Computational Intelligence and Industrial Application,vol 2 (2008), 1002-1005.

[6] H. Uozumin and Y. Shirai, Mobile Robot Motion Planning considering the Motion Uncertainty of Moving Obstacles, IEEE Int. Conf. on System, Man, and Cybernetics, 1999, 692-698.

[7] J. Kim Pearce and R. Amato, Extracting optimal paths from roadmaps for motion planning, procedding IEEE International Conference on Robotics and Automation, 2003.

[8] H. Li and SX. Yang, A behaviour-based mobile robot with a visual landmark recognition system, IEEE transactions on Mechatronics, vol 8 (2003), 390-400.

[9] D. Rodgriguez and L. Pedraza, ``Latest developments in feature based mapping and Navigation for indoor service robots'', Nova Science Publishers, 2009, 123-150.

[10] A. Zhu and S. Yang, ``An Adaptive Neuro Fuzzy Controller for Robot Navigation, `` Book chapter in Recent Advances in Intelligent Control Systems, Springer, 2009, 277-281.

[11] www.sourceforge.net/projects/opencv.

[12] J.F Canny, The complexity of robot motion planning, MIT Press, Cambridge, MA, 1988.

[13] T. Simeon and J.P Laumond, Manipulation planning with Probabilistic Roadmaps, International Journal in Robotics Research, vol. 23(2004), 729-746.

[14] E.W. Dijkstra, A note on two problems in connection with graphs, Numerical Math. I (1959), 269-271.

[15] Briggs et al, Expected Shortest Paths for Landmark-Based Robot Navigation, The International Journal of Robotics Research, vol. 23(2004), 717-728.

[16] J.-S. R. Jang, ANFIS: Adaptive-network-based fuzzy inference systems, IEEE Trans. Syst., Man, Cybernetics, vol. 23(1993), 665-685.

[17] J. -S. R. Jang, C.T. Sun, E. Mizutani, Neuro Fuzzy and Soft Computing, Prentice Hall Publisher, 1997.

[18] R. Petru and M. Emil, Behaviour-based neuro-fuzzy controller for mobile robot navigation, IEEE Transaction on Instrumentation and measurement, vol 52, no. 4(2003).

[19] D. Nauck and R. Kruse, Designing neuro-fuzzy systems through backpropagation, In Fuzy Modelling:Paradigms and Practice, Kluwer, Boston, 1996), 203-228.

[20] J.J. Kuffner, K. Nishiwaki, S. Kagami, M. Inaba and H. Inoue, Motion planning for humanoid robots, Proc. 20th Int'l Symp. Robotics Research (ISRR'03), 2003.

[21] B. Stanley and P. McKerrow Measuring range and bearing with a binaural ultrasonic sensor, Proc. of the IEEE/RSJ International Conf. on Intelligent Robots and Systems,vol.2 (1997), 565-571.

[22] Indian database, http://vis-www.cs.umass.edu/˜–˝vidit/ IndianFaceDatabase