The Implementation of Embedded Vision System for Line Follower Navigation

Razali M.T, Hafize Rahmat, Herdawati Abdul Kadir and Mohd Helmy Abdul Wahab

Abstract – Line follower is one of the important capability that mobile robot must have in order to navigate successfully in competition. Normally, robotic system employs infrared sensors for this purpose because of its simplicity and accuracy. The disadvantages of using this sensors is that width of the line must be known at the first place, color between the line and background must be contra enough and it is quite hard for infrared sensors to track back the line when the robot is out of track. In this paper, line follower navigation system using CMUCam2 was proposed. The camera is attached to small mobile robot for enhancing its ability to track the color line and eventually follow the line accordingly. The robot is able to distinguish different color line within camera's field of view and successfully navigate through the developed line track.

INTRODUCTION

The rapid growth in robotics field nowadays improves a lot of developments in many fields. Robot with vision ability enhances the accuracy and flexibility since its provide a lot of advantages such as robust path planning, flexible coordinate placing and etc. As to date, there are a lot of applications employed robots with vision system especially in manufacturing industry, research, science development and space navigation.

Vision is a powerful sensor because its ability to acquire a lot of data from the surrounding. The creation of an artificial vision system for a robot to see is one of the most difficult and challenging area in robotics field. Robot with a capability to see is not just simply connecting a camera to the robot. The images must first be represented and reconstructed in a way computer understand it before further analysis can be done. Once this stage completed, various algorithm can be design in order to meet the application requirement. Because of the process involved is complicated, normally computer was used to execute the algorithm. However computer based vision system use a lot of space and it is only fit to a big size robot. As robot become smaller nowadays, it is important to develop an embedded vision system that can be implementing by using microcontroller.

Vision system implemented using embedded system is quite challenging work. An associated controller is needed to interface to the frame grabber and executes the algorithm. The aim of embedded vision system is to provide the functionality in a small size and low power package and low data bandwidth stream to a host processor like microcontroller. The advancement of technology nowadays makes it possible to develop a high speed and performance, small size and low cost microprocessor.

Embedded vision system is a combination of embedded system and vision system. It can be divided into two major areas which are machine vision and computer vision [1]. These two systems were classified on how images were created and processed. Machine vision process image by increasing reliability and reducing cost of equipments and complexity of algorithms and normally used in manufacturing fields. This is oversimplified the situation compare to computer vision. Computer vision functionality is to process image of real world video or photography and mostly operated in human environments.

Embedded vision system prove to be extremely useful in variety of applications because of the benefits' offer such as sophisticated vision technology, savings cost, improved quality, reliability, safety and productivity. Quality control is one of the many applications of vision system that already implemented in industry. The main purpose of this application is for supporting manufacturing process in automatic quality control for inspection of final products in order to determine defect.

One of the largest fields for vision system is mobile robot navigation [2]. The examples of inventions are detection of enemy soldiers, mine exploration and surveillance system. Vision system technology provides information about surrounding area that can be used for path planning. Beside that, prominent application area is in autonomous unmanned vehicle [3], mini robot navigation [4] and soccer robot [5]. Typically, autonomous vehicles use vision system for navigation and the tasks can be simplified as below:

i) Producing a map of environments
ii) Detecting obstacles
iii) Looking for forest fire
iv) Obstacle warning systems in cars
v) Autonomous landing of aircraft
vi) Space exploration

The paper is organized as follows. Section 2 presents architecture of CMUCam2 and how to implement it in a real application. Section 4 describes in detail results for camera acquisition, processing and how to use the acquired information for controlling robot movement. The summary and future researches are presented in section 5.

SYSTEM DETAILS

The architecture for vision based line follower system as shown below:

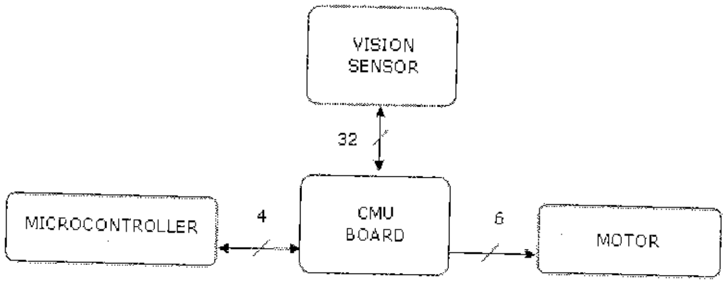


Figure 1: The pin connection and component involved.

There are four main parts that will be use in the design namely vision sensor, CMU board, microcontroller (PIC 16F876A) and servo motor. Vision sensor received a specification data of the object or line for tracking based on the color information. When the target is detected, image processing algorithm will be execute in the CMU board and result after the process finished will be send out to the microcontroller for further process. During program execution, every instruction that receive from the CMU board will be used to performed motor action either goes straight, turn left or turn right. The summary of the process is shown below in Figure 2.

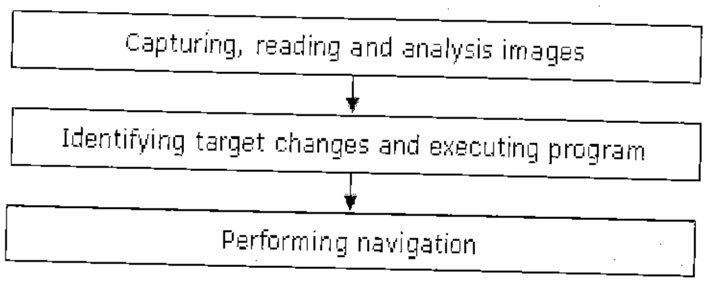
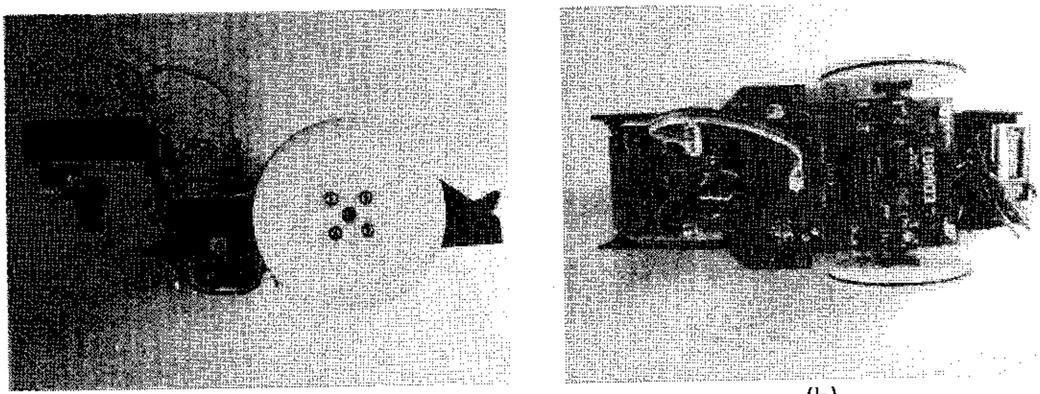


Figure 2: The process performs by the system.

Vision Sensor and CMU board.

Vision sensor for this system employed CMUCam2 v1.0 [6]. Its consist of two parts which are CMOS camera and CMU Vision Board. In order to enable the CMOS camera to trace target, the numerical value for each channel (Red, Green and Blue) must be set up. Channel is a part which generates different voltage due to the different amount of light passes through the lenses. Amount of light depends on target color. This system use value channel in range of 16 to 240 to detect certain object or line as a target

CMUCam2 Vision Board was used due its capability to enable onboard image processing algorithm and thus it does not need a high performance processor to execute the algorithm. It gives advantage to apply vision system for a small mobile robot and it is simple to use. The picture of mobile robot attached with CMUCam2 and the pin involve in this project is shown in Figure 3 and Table 1 respectively.



(a) (b)

Figure 3: Mobile robot attached with CMU board. (a) Side view. (b) Top view

Table 1: CMU board port assignment

PORT NAME	PIN	FUNCTION
Servo output	Output 2 (positive, negative and signal)	Left servomotor
	Output 4 (positive, negative and signal)	Right servomotor
Power	positive and negative terminal	+ 12 V
Servo power	positive and negative terminal	+ 5 V
Camera bus	32 pin	CMOS camera chip
TTL serial	Logic in (SRX)	Tx pin (microcontroller)
	Positive terminal	+ 5 V
	Logic out (SRX)	Rx pin (microcontroller)
	Ground	negative terminal

Microcontroller

PIC16F876A was used to control the process for reading and execute output command. Since SX52 microcontroller that attached to the CMU board is used to process the video data, thus there is no need for large memory space.

The main program flow will be kept in PIC16F876A to control the operation below:

i) Identify and track the color line.
ii) Acquire the coordinate of the line.
iii) Control motor to follow target line either turn left, turn right or move forward.

The entire programs were design and developed using MicroC and compile using MicroElektronika C compiler for Microchip PIC microcontroller. The PIC used 20 MHz Crystal and input and output pin used is shown below;

Table 2: Input and output microcontroller pin assignment

PIN NUMBER	PIN NAME	FUNCTION
1	MCLR	Reset
9	OSC 1	Osc
10	OSC 2	Osc
15	RC 4	Boot
17	RC 6	Transmit data
18	RC 7	Receive data
19	Vss	+ 5 V
20	Vdd	Ground

The CMU board allows controlling image processing algorithm by using 47 special commands. All algorithms used were executed using PIC16F876A that connected to the CMU board using serial protocol. In this project, the command used for line navigation consist of eight commands and shown with detail explanation in Table 3.

Table 3: CMU command syntax and its description

COMMAND SYNTAX	DESCRIPTION
RS	Camera reset to default mode
NF 2	Noise filtering
CR 18 44	Set camera's mode to RGB mode with white balance ON
BM 0	New frame is continuously being pushed into frame buffer
L1 2	Turn ON led on camera when detect color while tracking
TC 66 116 0 41 0 41	Track red color between color range R = 66, G=0, B=41 (for minimum RGB value) and R = 116, G = 0, B = 41(for maximum RGB value). After using this command camera will returning T packet data. This data then will manipulated by the system.
SV 0 255	Run servomotor for port number 2 with speed of 255
\r	Set camera to idle state. Syntax must be put at end of every camera's command syntax.

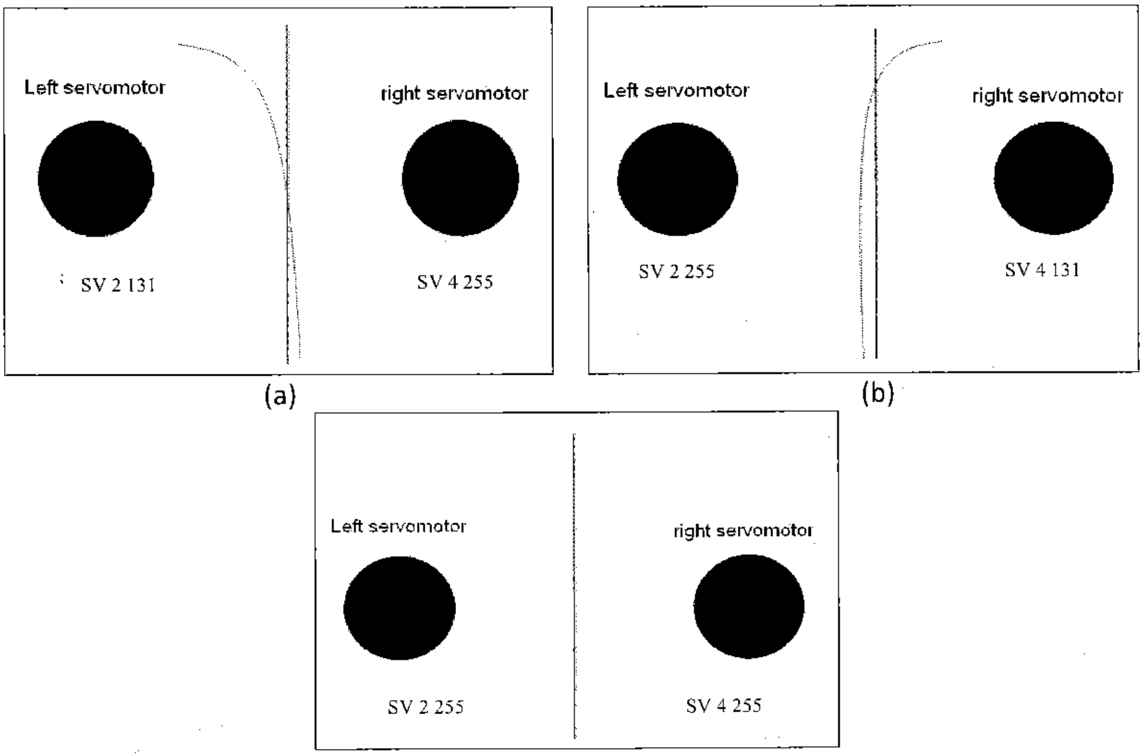
Motor

Motor is used for controlling the robot movement. The motor used in this system are servo motor from Parallax since it can continuously rotate and suitable for navigation purpose. Two servo motor are wired directly to the CMU main board and some of the important command to control the servo motor speed is shown in Table 4below.

Table 4: Motor control command

SERVO SPEED	DIRECTION
0 to 130	Counter clockwise
131	Stop
132 to 255	Clockwise

Figure 4 shows how to make the robot turn to left, right and forward direction using the built in CMU command.



(c)

Figure 4: CMU board servo command. (a) Turn left. (b) Turn Right. (c) Move forward.

RESULTS

Figure 5 shows the first experiment done to detect the red color and discard the rest. The dark blue line was put together with the red line to prove the effectiveness of the tracking color algorithm. In order to track the red color, the minimum and maximum RGB values must be set and the value was obtained experimentally. The suitable minimum and maximum values for each color channel acquired from the experiment is shown in Figure 8. The first 70 is a minimum value and 120 is a maximum value for red channel. The other values 0 and 41 are represented minimum and maximum values for green and blue channel respectively. If the acquired pixel value from the camera lies between 70 to 120 for red channel and 0 to 41 for green and blue channels, then the pixels will be set as a foreground, beyond that it is a background.

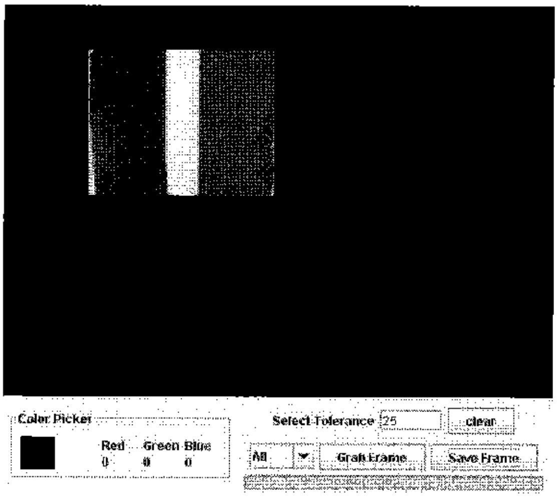


Figure 5: Picture tracing red and blue color by robot

Figure 6: The minimum and maximum RGB value

Finally, after tracing command (TC) is applied with values in Figure 6, the result obtains as shown in Figure 7. It's clearly seen that the red color was successfully detected without a noise. The light blue area means that tracked foreground and the grey area is a background. The red dot in light blue area represents the center of mass of the red color.

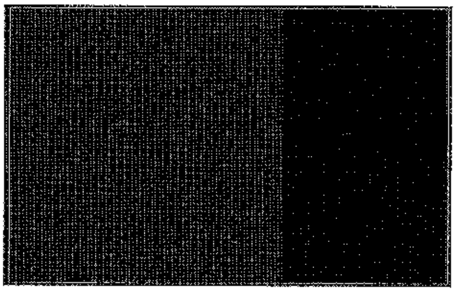


Figure 7: Result shows the red color is traced by using image from Figure 7

The second experiment was done to test the ability of the mobile robot for tracking the red line and navigate through the line successfully. Figure 8 shows the original picture acquires from the camera. If tracking color command is executed, a lot of noises were obtained along with the picture and thus not applicable for further process. As a solution, white balance was set in order to threshold the area of light color into white color by using CR 18 44 command, where 18 is color mode register and 44 is a values to turn on RGB auto white balance. The result after this process is shown in Figure 9. Next process is to enhance the image visibility by alter the camera focusing lens so that foreground area more visible than the background. Result before and after this process is shown in Figure 10 and Figure 11 respectively. After the filtering part was done, segmentation part will take action to prune out red area and eliminate white area using tracking color command. Image result for the segmentation process is shown in Figure 12.

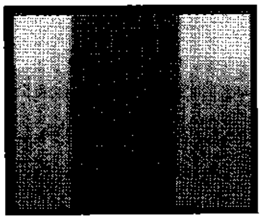


Figure 8: Original picture of the red line

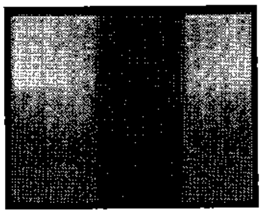


Figure 9: White balance is set using command CR 18 44

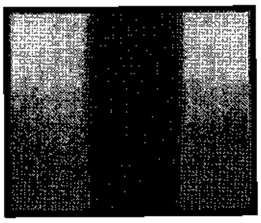


Figure 10: The red line picture before the camera is focus.

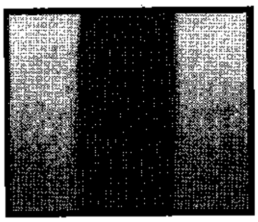


Figure 11: The red line picture before the camera is focus.

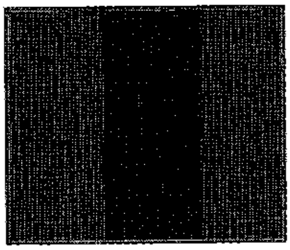


Figure 12: Red color is at the centre of Camera's field of view

After the segmentation process finished, an important features need to be extract from the segmented image for the purpose of navigation. This project use T data packet information that was obtained after tracking command was executed. Details explanation about T data packet is shown in Table 5.

Table 5: T data packet

	T mx my x1 y1 x2 y2 pixels confidence
mx	Middle of mass x value
my	Middle of mass y value
x1	The left most corner's x value
y1	The left most corner's y value
x2	The right most corner's x value
y2	The right most corner's y value
pixels	Number of Pixels in the tracked region, scaled and capped at 255 : (pixels +4)/8
confidence	The (# of pixels / area)*256 of the bounded rectangle and capped at 255

T data packet need to be manipulated in order to perform the navigation task. When the red line is in the middle, leftmost and rightmost camera's field of view the return T data packet is shown in Figure 16, Figure 17 and Figure 18 respectively. From these figures, the important parameter that used for navigation is middle of mass x value (red circle) since this value able to give roughly idea current orientation of the robot and how to predict the movement of the mobile robot.

```
T 47 66 29 1 64 141 255 225
T 47 67 29 1 64 141 255 227
T 47 66 29 1 64 141 255 224
T 47 66 29 1 64 141 255 225
```

Figure 16: T data packet of Red color at centre of camera's field of view

```
T 4 50 3 1 7 120 39 134
T 4 50 3 1 7 127 39 125
T 4 49 3 1 7 127 39 124
T 4 50 3 1 7 125 39 127
```

Figure 17: T data packet of Red color at left most corner of camera's field of view

```
T 86 47 84 1 87 114 30 134
T 86 48 84 1 87 115 31 136
T 86 45 84 1 87 106 30 142
T 86 46 84 1 87 116 29 129
```

Figure 18: T data packet of Red color at right most corner of camera's field of view

The algorithm for navigation can be summarized as a pseudo code shown below. :

INITIALIZE CMUCAM2
TRACKING COLOR
GET MIDDLE OF MASS X VALUE
IF middle of mass x value > 47
THEN turn right
ELSEIF middle of mass x value < 47
THEN turn left
ELSE move forward

This algorithm was successfully implemented to navigate the mobile robot through the line track and able to make the robot move accordingly with only information from the middle mass x value.

CONCLUSION

In this paper, a systematic method for implementing embedded vision system for line follower navigation was discussed. The systems successfully detect target color and concurrently navigate the robot to follow the line. From the experiments, it is conclude that under bright condition and constant ambient light a good performance can be achieved.

In future, the system can be improve by design more complicated algorithm for high speed navigation and handle with multiple line sizes.

REFERENCES

[1] A. Aggarwal, "Embedded Vision System (EVS)", IEEE/ASME International Conference on Mechatronic and Embedded System and Application, Beijing, China, Oct. 12 – 15, 2008.

[2] N. Sawasaki, M. Nakano, Y. Yamamoto and K. Okabayashi, "Embedded Vision System for Mobile Robot Navigation", IEEE International Conference on Robotics and Automation, Orlando, Florida, May 15 – 19, 2006.

[3] J.D Anderson, Dah-jye Lee, B. Edwards, J. Archibald and C.R. Greco, "Real-time Feature tracking on an Embedded Vision Sensor for Small Vision-guided Unmanned Vehicles", International Symposium on Computational Intelligence in Robotics and Automation, Jacksonville, Florida, June 20 – 23, 2007.

[4] J. Palacin, A. Sanuy, X. Clua, G. Chapinal, S. Bota, M. Moreno and A. Herms, "Autonomous mobile mini-robot with embedded CMOS Vision System", IEEE 2002 28th Annual Conference of The Industrial Electronics Society, Nov. 5 – 8, 2002.

[5] A. Sluzek and P.K.D. Minh, "Embedded Vision Module for Robot-Soccer", IEEE International Conference on Computer Systems and Application, March 8, 2006

[6] A. Rowe, C. Rosenberg nd I. Nourbakhsh, "A Second Generation Low Cost Embedded Color Vision System", IEEE Computer Society Converence on Computer Vision and Pattern Recognition, San Diego, California, USA, June 25 – 25, 2005.