TRACK AND OBSTACLE DETECTION WITH USB CAMERA FOR VISION-BASED AUTOMATED GUIDED VEHICLE

Sulaiman Sabikan*, Prof. Dr. Marizan Sulaiman, Syed Najib Syed Salim, Muhammad Fahmi Miskon

Abstract: This paper describes a low cost tracking system and obstacle detection with Universal Serial Bus (USB) camera for Vision-based Automated Guided Vehicle (V-AGV). By using Machine Vision Software, the visual information obtained from the USB camera will be processed using image processing techniques such as Region of Interest, smoothing and edge detection for detecting tracks and obstacle(s) that might appear on the navigational track system. Algorithms that can be used for V-AGV navigation without collision are suggested. The distance between V-AGV and obstacle, the size, and height of the obstacle can also be measured using a single USB camera. Using the advantages available in the software, the results from some experiments are presented.


1.0 INTRODUCTION

Automated Guided Vehicle (AGV), one of one kind of transportation units that follow perspective paths and routes is widely used in industrial field, community service and in dangerous working area. In the development of AGV, there are two main navigation systems, guiding with lines and without lines. The AGV guided by lines can operate with simple design and control. The cable magnetic tape guided is most popular because of reliability and accuracy. However, it costs a lot in introduction and maintenance of the system, but being weak at changing the path and relocating the job. Other navigation system is without guided line such as Beacon System and Dead Reckoning has well flexibility, but the former is complicated and has internecine, the latter has a problem of error accumulation. [1][2]. Most AGV’s use various sensors such as ultrasonic sensors, infrared sensors, limit (bumper) sensors, magnetic sensors, vision system and speech recognition to accomplish their task in various circumstances. It incurs higher price on an AGV system and requires a lot of maintenance. Besides that, most systems that utilizes guided designs such as magnetic lines, requires greater maintenance and cost.

If the system is equipped with computer image processing or computer vision system and introducing low cost guided lines, it can make the design of the guiding line flexible, the guiding mark recognition easier, system operation robust and reduce the overall cost.

* Faculty of Electrical Engineering, Universiti Teknikal Malaysia Melaka Locked Bag 1200, Hang Tuah Jaya, Ayer Keroh, 75450, Melaka. Email: sulaiman@utem.edu.my
In the development of AGV, line detection and recognition methods by image processing techniques also came to interest. Wavelet transforms and Hough Transforms are most popular for this purpose [2]. In many researches in vision-based area, a lot of techniques were introduced in its navigation system such as feedback optimal controller [3], fuzzy logic controller [4], neural network and interactive learning navigation.

With the fast development of machine vision system and advanced computer technology, system that implements vision has many advantages compared to other systems that use different type of sensors [2]. It includes low cost in introduction and maintenance, the better and larger amount of information extracted from a single image.

This paper introduces the application of Machine Vision Software in developing track and obstacle detection for AGV’s with fast execution, high reliability and only requires minor modifications for adjustments to be used on any platform. It also introduced low cost Universal Serial Bus (USB) camera as a single input sensor.

This paper is organized as follows. The configuration of AGV is introduced in Section II. Section III and Section IV details the track and obstacle detection algorithm respectively. Section V shows the experiment result and lastly Section VI is conclusions.

2.0 CONFIGURATION OF AUTONOMOUS GUIDED VEHICLE

The configuration of AGV discussed in this paper includes, an ER1 Personal Robot as a platform, Machine Vision Software, a USB camera and a laptop computer. The ER1 Personal Robot is a commercial mobile robot system that includes the control software, motor driver, aluminum beams and plastic connectors to build a chassis, two assembled nonholonomic scooter wheels powered by two stepper motors, one 360 degree rotating caster wheel, a power module and a battery (12V 5.4A). A laptop computer, Toshiba Protégé (1.2GHz, 256MB of RAM), is used as a controller device, and Windows XP Professional is loaded as the operating system. The Halcon 7.0 Machine Vision system is used as image processing software and Visual Basic High Level Language will be implemented to integrated Image processing software, USB camera driver and the robot controller software.

The USB camera install at the top of AGV and pointing ahead to get the forthcoming information of the lane, environment perception such as obstacles and special sign identification.

3.0 TRACK DETECTION ALGORITHM

The V-AGV requires a guiding system for navigation, with or without lines. By combining with the vision system, the V-AGV is more flexible, the guiding marks recognition easier, and more robust. The advantage of machine vision system used speeds up the project development, adds flexibility and in total produces a more powerful system.
In this project, the design of the AGV’s track is shown in Fig. 1. The width of the track is fixed at 1.7 cm using black cello tape with a background of white cello tape with a width of 4.8 cm. This makes the preparations for the system incur very low cost and easy maintenance.

![ROI 1, ROI 2 and ROI 3](image)

Fig. 1: Region Of Interest (ROI 1, ROI 2 and ROI 3)

The first problem that must be solved is the real time image acquisition direct from the USB camera without using any interface with frame grabber. The immediate effect is the hardware for the V-AGV is low in cost and easy to obtain. Any type or brand of USB camera can be used in this project. The Kriter USB camera was used in this project. A problem encountered is that the image obtained is of low quality and needs a few existing image processing techniques suitable to increase the image quality.

The image acquired is in low quality, with a size of 320x240 pixels and in gray value scale format. The linear smoothing technique by averaging with grey values of all input image is used for border treatment at the image edge by using \textit{mean\_image} operator. The operator \textit{mean\_image} carries out a linear smoothing with the gray values of all input images from USB camera. The filter matrix consists of ones (evaluated equally) and has the size \textit{MaskHeight x MaskWidth}. The result of the convolution is divided by \textit{MaskHeight x MaskWidth}. \textit{MaskHeight} and \textit{MaskWidth} is set with the value 3 pixels. By using this technique, noise on the image can be reduced. Fig 2 below shows the result before linear smoothing on the left and after linear smoothing on the right.

![Before and After Linear Smoothing](image)

Fig. 2: Before and After Linear Smoothing
Using only the fixed ROI’s, Track Detection Algorithm uses the command `measure_pos` to determine the existence of the track in a processed image. This operator combines two important processes, extracts straight edges which lie perpendicular to the major axis of a rectangle of ROI and measure distance between consecutive edge points. This process is only applied in the ROI 1, ROI 2 and ROI 3 with the intention of speeding up image processing and avoid time wasting. The distance value of the track has to be between black cello tapes with a background of white cello tape. As a result of this process, the coordinate edges value \((RowEdge,ColumnEdge)\) and the track width \((Distance)\) in all ROI’s can be obtained. With the presence of a white background track, the vision system can recognize the black track on a floor of any color.

Referring to Fig. 3, there are two parameters marked red. It refers to the center point on ROI 1, Coordinate 1, \(U(x,y)\) on top and center point of ROI 3 Coordinate 2, \(S(x,y)\) at the nearest point closest to the V-AGV platform. The main parameter is the pixels Coordinate 2, \(S(x,y)\). It is obtained from the following formula.

\[
\text{if (} (DistanceS[j] >= TWidSmin \text{ and } DistanceS[j] < TWidSmax \text{ and } AmplitudeS[j]<0 \text{ and } AmplitudeS[j+1]>0) \text{) so } \\
S=[ 192.0, ColumnEdgeS[j+1]-(DistanceS[j]/2)]
\]

Where \(ColumnEdgeS\) and \(DistanceS\) is obtained from the results of the image processing step while other parameters are fixed.

The value 192.0 is the row value for the centre of ROI 3.

Furthermore, the value of pixels Coordinate 1, \(U(x,y)\) can be obtained from the following formula.

\[
\text{if (} (DistanceU[j] >= TWidSmin \text{ and } DistanceU[j] < TWidUmax \text{ and } AmplitudeU[j]<0 \text{ and } AmplitudeU[j+1]>0) \text{) so } \\
U=[ 192.0, ColumnEdgeU[j+1]-(DistanceU[j]/2)]
\]
Where ColumnEdgeU and DistanceU is obtained from the results of the image processing while other parameters are initially fixed.

The value 48.0 is the row value for the centre of ROI 1.

In the Fig. 3, Coordinate 1 and Coordinate 2 are U[48.0,155.9] dan S[192.0, 157.9] respectively.

![Fig. 3: Track Detection, Coordinates U and S.](image)

Based on the two main coordinates, other parameters can be calculated with simple formulas such as track distance from the image centre position, distance from V-AGV and track angle. The centre position of the image is fixed.

### 4.0 OBSTACLE DETECTION ALGORITHM

In this section a simple method is used to detect an obstacle on the track while the AGV is moving. An obstacle on the track is seen by the USB camera. This paper proposes a simple approach to detect an obstacle on the track. In any case we consider the obstacle appears if track detection image processing cannot detect any value of pixels which are coordinates U and S.

When the obstacle is sensed, AGV will record data such as the width of an obstacle, and will identify the types of obstacle either cubic or flat ones. Thus, it will then stop and analyzes the object located in front of it. Two variables, Distance 1 dan Distance 2 is obtained from the image from two Region of Interest, ROI 1 and ROI 3. The difference of these values determines whether the obstacle is cubic (with height) or just a flat obstacle. Referring to Fig 4 and Fig 5, if the pixel value of Distance 1 is greater than Distance 2, the V-AGV will know that the obstacle in front is cubic with a certain
height. Otherwise, if Distance 2 is greater than Distance 1, the shape of the obstacle is flat. The ability to detect the obstacle type enables the V-AGV to plan the right course of action to avoid the obstacle in front of it.

Fig. 4 : Cubic Obstacle

Fig. 5 : Flat Obstacle

5.0 RESULT

Experiments have been done on Vision-Based Navigation System to detect track and obstacle and it functions well. In this experiment, we develop a tracking sistem in a controlled laboratory room in terms of environment and lighting source. There are a few guidelines built such as straight-line, crossing-line and also simulated broken line. Besides that, we also provided a few obstacles in a number of places along the path of the AGV to test the obstacle detection algorithm developed. In this paper a few examples of obstacles is used which includes the big box obstacle and flat obstacles. In
this experiment results section, it is divided into two segments; track detection algorithm and obstacle detection algorithm.

Throughout the experiment, image processing successfully processes all image guidelines obtained by the USB camera including tracks that has problems. Fig. 6 refers to the data shown in the screen as a result of the track detection algorithm while AGV following a straight line, a crossing line and a broken line. The data obtained are Pixels of Coordinate 1 and 2, the type of track, image processing time, track angle with respect to the image centre, track distance from image centre to Coordinate 1 and 2.

Fig. 6 : Straight line, crossing line and broken line data
According to the screen in Fig. 6, this information can be obtained;

Straight line
U=[x,y]=[48.0, 122.9] pixels
S=[x,y]=[192.0,154.2] pixels
Type of Track = Straight
Image Processing Time: 79ms
Track angle = 11.53 deg
Track distance for ROI 1 = -33.97 pixels
Track distance for ROI 3 = -5.75 pixels

Crossing line
U=[x,y]=[48.0, 150.5] pixels
S=[x,y]=[192.0,154.8] pixels
Type of Track = Straight and Left
Image Processing Time: 45ms
Track angle = 0.96 deg
Track distance for ROI 1 = -6.40 pixels
Track distance for ROI 3 = -5.22 pixels

Broken line
U=[x,y]=[48.0, 170.1] pixels
S=[x,y]=[192.0,1172.2] pixels
Type of Track = Straight
Image Processing Time: 156ms
Track angle = 0.07 deg
Track distance for ROI 1 = 13.16 pixels
Track distance for ROI 3 = 12.17 pixels

Besides that, through the experiment, two obstacles were placed in the AGV’s path while moving. When this is done, the AGV stops and identifies the obstacles ahead. After the obstacles are removed, the AGV will continue its movement as usual.

Fig. 7 below shows the experiment results on the obstacle detection introduced. The image on the top figure in Fig. 7 indicates that the AGV can analyze the obstacle width in to parallel locations and identifies that it has a height exceeding the AGV’s height while the image on the bottom figure is a flat obstacle placed on the AGV’s path.
The values D1 and D2 obtained from Halcon Output data variable is shown in Fig. 8 below. The results, value D1 is 174 pix and D2 is 164 pix and by using the obstacle algorithm formula, the AGV successfully identifies the obstacle in front with a height greater than the AGV itself. The result at the bottom of Fig.8 also shows D1 value as 125 pix and D2 as 181 pix. This shows that D2 is greater than D1. Furthermore, the AGV also successfully detects the presence of an obstacle in front of it and it is lower than the AGV itself.
6.0 CONCLUSION

This paper has given tally description of a Vision-Based navigation system for an AGV. Using ERI Commercial Robot Kit as the platform, image taken from USB camera is analyzed using certain techniques of image processing. Introducing USB camera in the system with the applied technique, although the image obtained would have lower quality effect, is sufficient to trace back the prepared track and gaining other information that is crucial such as tracing back crossing track, intermittent track and tracing back the existence of obstacle in the path. Next, the data obtained from the image can be analyzed by using track detection algorithm that has been proposed to get the current position and AGV orientation that enables AGV to follow the centre line on the track. Using this method, AGV can move to follow the track successfully. Also, by using single sensor (USB camera), the AGV can detect and measure the width of obstacle and can stop to prevent collision. Further study regarding this paper is still in progress, which focuses on Navigation Intelligent system including speed controller, path planning strategy and collision avoidance algorithm.

7.0 ACKNOWLEDGEMENT

I wish to express my gratitude to Universiti Teknikal Malaysia because this project was sponsored by IRPA FKE KUTKM (now UTeM) research grant.
8.0 REFERENCES


