Easy *i*-Move: Structured Image Recognition Solution for Automatic Guided Vehicles

Yen-Hui Kuo[1,2](✉), Shu-Hui Lin[1,2](✉),
and Eric Hsiao-Kuang Wu[2](✉)

[1] Institute for Information Industry, National Central University,
Taoyuan, Jhongli, Taiwan
{yenhui,vickylin}@iii.org.tw
[2] Department of Computer Science and Information Engineering,
National Central University, Taoyuan, Jhongli, Taiwan
hsiao@csie.ncu.edu.tw

Abstract. There are many kinds of image processing application in the industry, to improve better performance of producing, Automatic Guide vehicles (AGV) solution have been driven in service, for instance, the expensive solution such as light detection and ranging (LIDAR), magnetic tape guided and RFID are used. In the industry which not only search for better delivery performance but also search for economic efficiency. Here we present an economic efficiency solution, Easy *i*-Move for AGVs guiding system to fulfill customers' necessity. In this study, we focus on color tape image recognition technology based on HSI color model. The implementation of the system architecture and the technology of image recognition in the proposed system are described and validated in this paper.

Keywords: Image recognition · Automatic Guided Vehicle · Color model

1 Introduction

There are many kinds of d Guide vehicles guiding solution, for instance, the light detection and ranging (LIDAR), a remote sensing method used to examine the surface of the earth, and magnetic tape guided are used. In the industry section which not only search for better delivery performance but also search for economic efficiency. Usually, manufacture working area reorganized from time to time. Working area reorganized which caused a very much housekeeping job to do subsequently. When reorganization occurs, the course needs to change. Therefore, we are trying to apply the color tape image recognition AGVs guiding strategy to handle such situations. There are some problems exist when we apply image recognition solution for AGVs guidance. We have to solve (1) Image processing time must short enough for guiding car:

The most important things in the Automatic Guide Vehicles guiding system is guiding response time. Car is running in real time, about 50 cm per second, image recognition system must take a quick response to guiding control module, which generates control instruction to AGV car. (2) Lighting noise proof: In the manufacturing there are a lot of lighting noise, such as the light from fluorescent, LED light

J.-S. Pan et al. (eds.), *Advances in Smart Vehicular Technology, Transportation, Communication and Applications*, Smart Innovation, Systems and Technologies 86,
https://doi.org/10.1007/978-3-319-70730-3_1

from instruments, sun light from outside etc., the image recognition algorithm not only needs to quick response but also need to analyze the picture accurately.

Because of the previously stated concerns, this paper will propose an AGV system using image recognition technology. We implement an AGVs system to verify this solution.

The rest of this paper is organized as follows. In Sect. 2, the RGB and HSI color model is introduced. In Sect. 3, the challenges of color tape image recognition solution for AGV system will be discussed. In Sect. 4, the Image recognition strategy for AGV is exploited. In Sect. 5 deployment of the new intelligent AGV system are presented. We conclude the paper in Sect. 6.

2 RGB and HSI Color Model

2.1 RGB Color Model

The RGB color model is an additive color model in which red, green and blue light are added together. The name of the model comes from the initials of the three primary colors, red, green and blue. Initially, we used the RGB color model to implement image recognition system. Our development platform is windows visual studio using Emgucv library [1, 2], in c# language, we get RGB color attributes from a picture of each pixel though GetPixel(x, y) method, which return R, G, B value for red, green, and blue respectively, which value vary from 0–255. This method seems very straight forward to tell a specific color tape apart from a picture which we got from camera, but The AGV System running in manufacturing which environment is complicated, there are a lot of lighting noise as mention before, I cause the picture catch from the camera has color distortion. The ground color varies from grey to blue, as shown in Fig. 1.

normal color distorted color

Fig. 1. Color distorted by light (Color figure online)

Initially, We used the blue color tape as guiding line, that color distortion like this causes a lot of trouble on image recognition, we can't tell part the guiding line from ground simply use R, G and B value from RGB color model, so we try HSI color model solution [3].

2.2 HSI Color Model

The HSI color model represents every color with three components: hue (H), saturation (S), intensity (I). When humans view a color object it is described by its hue, saturation and brightness. Hue is a color attribute that describes a pure color (pure green, blue or red). Saturation gives a measure of the degree to which a pure color is diluted by white light. Intensity is key factor in describing color sensation. We convert RGB color space to HSI color space by using the following formulas [4–6].

$$\mathrm{numerator} = [(\mathrm{R}-\mathrm{G}) + (\mathrm{R}-\mathrm{B})] \div 2$$

$$\mathrm{denominator} \ = \sqrt{(\mathrm{R}\ \ -\mathrm{G})^2 + (\mathrm{R}-\mathrm{B})(\mathrm{G}-\mathrm{B})}$$

$$\left\{\begin{array}{l} \mathit{if}(\mathrm{denominator}) \neq 0 \\ \quad \mathrm{Theta} = \mathrm{Acos}(\mathrm{numerator} \div \mathrm{denominator}) \times 180 \div 3.14 \\ \quad \mathit{if}(B\ <\ G) \\ \quad\quad \mathit{Hue} = \mathrm{Round}(\mathrm{Theta}, 2) \\ \quad \mathit{else} \\ \quad\quad \mathit{Hue} = \mathrm{Round}((1-\mathrm{Theta}), 2) \\ \quad \mathit{else} \\ \quad\quad\quad \mathit{Hue} = 0 \end{array}\right.$$

$$\left\{\begin{array}{l} \mathrm{if}(\mathrm{R}+\mathrm{G}+\mathrm{B}) \neq 0 \\ \quad S = 1 - [3 \times \min(\mathrm{R,G,B})] \div (\mathrm{R}+\mathrm{G}+\mathrm{B}) \\ \mathrm{else} \\ \quad S = -1.0 \end{array}\right.$$

$$\mathrm{I} = (\mathrm{R}+\mathrm{G}+\mathrm{B}) \div 3$$

In c# language, we get RGB color attributes from a picture of each pixel though GetPixel(x, y) method, which return R, G and B. We feed R, G, B value into above formula then can get H, S, I value easily. The value of Hue varies from −180–180, Saturation varies from 0–1, Intensity varies from 0–255, that gives much space to justify pixel color. We will exploit more detail in the fourth section. The Challenges of color tape image recognition solution for AGV system.

3 System Operation Components

AGV car consists of a front end camera, image recognition module and driving control unit. The front end camera captures the picture of course while running, image recognition module analyzes the colored tape position and based on the color tape position generates a direction calibration value or gives a reach the anchor point instruction to drive the AGV car.

3.1 Challenges

The image recognition module plays the major game in driving AGV car, image recognition module must give an accurate instruction for AGV car. There are implementation challenges involving potential lighting noise, image recognition errors, image analyze processing speed, which will cause AGV driving malfunction.

With the HSI color model technology, we get the maximum image recognition accuracy, proposed an image down size solution to speed up system performance, every driving control cycle keep within 50 ms, to meet the car stable speed on 50 cm per second of our system. Our proposed scheme not only maintains the driving security, but it also maximizes image recognition accuracy and stable driving speed.

4 The Image Recognition Strategy for AGV System

The Image recognition strategy for AGV System is based on color recognition of each pixel [7]. The Image recognition process is shown as Fig. 2.

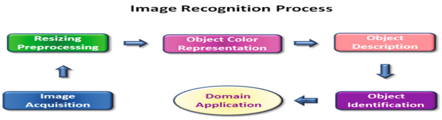


Fig. 2. Image recognition process

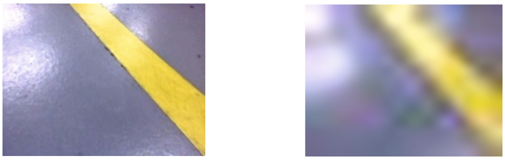
We get an image from camera, down resizing, read pixel color, telling object color apart from others, identify the right object, finally, output control instructions. We get the a picture from the camera consists of 640 × 480 pixels, if we used the raw picture to do Image analyze processing, it takes about 195.07 ms per picture, it takes too much time in processing the image analysis, it can't meet the AGV running speed requirement, it must less than 50 ms per analysis operation, we down size the picture to 20 × 20 pixels, it takes about 0.51 ms per picture, in our system the average operation time from take a picture to Image analyze and give an instruction to AGV car, it takes 2.08 ms from our 2002 sample pictures. The size of picture affected the Image analyze processing time shown in Table 1.

Table 1. Image analyze processing time

Platform	Image size	Analysis time IV (millisecond)	Meet requirement (50 ms<)
Intel Core i7-3540 M CPU: 3 GHz RAM: 8 GB OS: windows7	640 × 480	=195.07	✗
	100 × 100	=6.35	✓
	20 × 20	=0.51	✓

Our system must function normally despite the fact that the manufacturing, warehousing lighting noise. Furthermore, we want to maximize the system's accuracy as much as possible. Order to meet this requirement, we propose a HSI color model technology based on tape color. The tape color could be affected by a wide range of factors, such as fluorescent or LED color, the light from machine or equipment, and any lighting source. Accordingly, an accurate and reliable algorithm for Image analysis, the tape color must take into account the effects of these factors.

We exploit the detail of the image recognition algorithm, which we implement in our system. In the experimental manufacture, there is yellow color tape already, so we use yellow color for image recognition. After we captured a picture from end camera, we down size the picture from 640 × 480 pixel to 20 × 20 pixel, the picture shown as Fig. 3.



Original picture down size picture

Fig. 3. Picture down size before and after (Color figure online)

After the picture downsizing, it become blurred like the above right side picture, although the image becomes blurred, it is still good for recognition, and the most important is that the Image analyze processing time is short enough, so in the experiment 20 × 20 pixel is a good size for recognition. Based on Sect. 2.2 HSI color model formula, we caught each H, S, and I value of pixel[x, y] and saved in Hue[x, y], Sat[x, y], and Inten[x, y] array respectively. Here I draw 20 × 20 circle to represent the 20 × 20 pixel, as shown in Fig. 4.

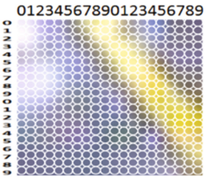


Fig. 4. 20 × 20 pixel picture

In real environment only H, S, and I value is not enough to tell similar pixel apart. Therefore, we propose additional HSI formula:

$$\mathrm{HSI} = \mathrm{H} \times \mathrm{S} \times \mathrm{I}$$

We further multiple the H, S, and I value generate a much more range and unique value to ensure color representation. With HSI model produces a representation of HSI, H, S, and I value. The Hue values of yellow color are positive around 50–60. In this example, we are lucky, we can tell the yellow color apart from Hue attribute as f(HSI,H) function shown.

$$f(\mathrm{HSI,H})\begin{cases} if\ HSI > HSI_{min} \\ \quad \text{then key factor is} \\ \quad H_{min} < H < H_{max} \\ \qquad where \\ \quad HSI_{\ min} = 1400 \\ \qquad H_{min} = 30 \\ \qquad H_{max} = 80 \end{cases}$$

In some other complicated case we have to use more attributes to tell them apart, as g (HSI, H,S,I) function shown.

$$g\ (\mathrm{HSI,H,S,I})\begin{cases} \text{if} \quad 0 < HSI \leqq \mathrm{HSI}_{min} \\ then\ \ key\ factors\ are \\ \quad H_{min} < H < H_{max} \\ \quad \mathrm{S} > S_{min} \\ \quad \mathrm{I} > I_{min} \\ \quad where \quad \mathrm{HSI_{min}} \quad = 1400 \\ A{:}\,H_{min} = 30,\ \ H_{max} = 50, S_{min} = 0.15, I_{min} = 100 \\ B{:}\,H_{min} = 50,\ \ H_{max} = 65, S_{min} = 0.05, I_{min} = 100 \end{cases}$$

Based on f(HSI,H) and g(HSI,H,S,I) function to represent the pixel color, but that can't descript the pixel is right in the line or caused by environment lighting noise, so our recognition algorithm has to add on line characteristics to verify line pixel. Both the side by side pixels of a line are not apart from 2 or more pixels, they must close together, nevertheless left or right, up and down. Apply this line characteristics we can descript the line image object, which

Cnt: Total chosen color count of row
Line pixel: (LineRightPos- LineLeftPos) ÷ Cnt < 1
Noise: (LineRightPos-LineLeftPos) ÷ Cnt ≧ 1

On behalf of the above rules, we get rid of some light noise from environment lighting or instruments light successfully.

Before we apply this solution in the system, we capture the whole course image at the stable speed say 50 cm per second, there are more than 10,000 pictures, and evaluated every misrecognized image carefully, then modify the parameters of f(HSI,H) and g(HSI,H,S,I) functions to make function more accurately.

We get the image structure of Fig. 4 With f(HSI,H) and g(HSI,H,S,I) functions as shown in Table 2.

Table 2. Image structure

Y	XI	X2	Line width
0	7	9	3
1	8	10	3
2	8	11	4
3	9	12	4
4	9	12	4
5	10	13	4
6	11	15	5
7	11	15	5
8	12	16	5
9	13	17	5
10	13	18	6
11	14	19	6
12	15	19	5
13	15	19	5
14	16	19	4
15	17	19	3
16	17	19	3
17	18	19	2
18	19	19	1
19	−1	−1	0

y: row number
x1: tape left side position
x2: tape right side position
line width: tape width

From Table 2 we mapped with the yellow tape image as Fig. 5. We can get a perfect pixel structure of yellow color matched with image picture.

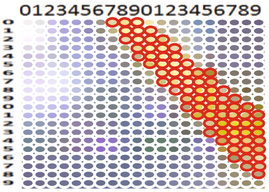


Fig. 5. Image structure data mapping (Color figure online)

5 Application of Image Recognition

At this point, we transfer the image picture to a structure of data table. How can we use these data for the application of AGV system? From the yellow color position in the 20 × 20 structure, we can conduct the car to follow. Before we use the yellow line position data, we have to make sure the image structure data is yellow line structure not from light noise or something else. In my algorithm, after we got the line image structure, we check the line length and line width first, if the line width great than 0 pixel and line length is great than 9 pixels, then we can sure it is a conduct line image, which we can use to conduct the car. There is a rule of the guiding line as below function.

Guiding line rules:

- Anchor line: line width > 16 and line length < 9; car is approaching the workstation
- Guiding line type1: line width > 0 and line length > 9; guiding line is located at the middle of picture
- Guiding line type2: line width > 0 and 3 < line length $\leq$ 9; guiding line is located at the corner of picture
- Guiding line type3: line width > 0 and 0 < line length $\leq$ 3; guiding line is located at the corner of picture.

From the above rules we identify the line type then we check the column position of the guiding line at the top, based on the position of guiding line to give the direction instruction. The guiding line can be set at the middle of lane or at the side of lane, the rule of at the middle of lane is:

(1) Forward: 7 ≦ guiding line top position ≦ 13
(2) Left: 0 < guiding line top position < 7
(3) Right: 13 < guiding line top position

Next, we carried out experiments in a real environment to verify the usability of the AGV system. Here, we implemented our system on a manufacture, which produces geared motors in Taiwan. We use an AGV car with a front end camera, so this AGV is implemented forwarding only, the driving instructions are send from recognition module to AGV control unit through USB port, the control unit is an Arduino module [8–10]. In the manufacture there are 15 Assembly stations in the manufacture, with a single route length of 180 m, about 6 min per circle. To investigate the performance of the proposed AGV system, we use the real manufacture lay out road map in the experiments.

Initially, AGV car is parked at the AGV parking lot, user asks AGV car to deliver products or materials from any assembly station to any station. After the car finish the delivery job, which will go back to parking lots, and check is there any job waiting for it.

6 Conclusions

In this experiment we take care of the Image processing time to meet the system requirement which around 0.5 ms on our platform. We also get rid of the lighting noise from outside to make the AGV system work in manufactory. There are many kinds of image recognition approach and application, some of further research applies machine learning or deep learning with image recognition technology to enhance image recognition system to specific domain application. Furthermore, image processing can be applied to many domain usages such as: satellite imagery, medical image, photo reconstruction, Automatic identification, defect detection, face discrimination, virtual reality, there are different domain knowledge inside. Here we have demonstrated the feasibility and usability of the proposed AGV system. After further investigation of image recognition solution performance, it will be possible to redesign the system architecture to more secure traffic control system. Many more new applications will then be developed in the near future.

References

1. Emgu CV Essentials. Packt Publishing, 14 November 2013
2. Emgu.CV.example CameraCapture. http://me1237guy.pixnet.net/blog/post/61361335
3. Converting from RGB to HSV. http://coecsl.ece.illinois.edu/ge423/spring05/group8/finalproject/hsv_writeup.pdf
4. Video color conversion. Introduction to Automotive Electronic Systems pdf. http://140.117.156.238/course/IAE/2013/
5. Gonzalez, R.C., Woods, R.E.: Digital Image Processing, 3rd edn. Prentice Hall, Upper Saddle River (2008)
6. Setting up Emgu CV Project with Visual Studio. https://notebookbft.wordpress.com/2015/03/08/setting-up-emgu-cv-project-with-visual-studio/
7. The Principle of Image Processing Technology and Application. http://web.ncyu.edu.tw/∼lanjc/lesson/C9/class/11.pdf
8. Arduino Mini USB Adapter. http://elesson.tc.edu.tw/md221/pluginfile.php/4151/mod_resource/content/1/arduino.pdf
9. Online electricity knowledge. http://bbs.audiohall.net/viewtopic.php?t=1337&sid=999
10. http://elesson.tc.edu.tw/md221/pluginfile.php/4151/mod_resource/content/1/arduino.pdf