

Design and implementation of automatic sorting car based on OpenMV

*Lin Zhu, and Yan Yang**

School of Computer and Information Engineering, Beijing Agricultural College, China

Abstract. The system uses the OpenMV algorithm module combined with the k-means framework through the Euclidean distance formula to facilitate the automation of logistics sorting, which identifies the color of logistics parcels in the LAB color space. It sends the identified color information to the motion drive module, which sorts and transports the logistics parcels to the designated storage area. The fusion of algorithms allows the system to identify colors and transportation accurately.

1 Introduction

The development trend for intelligent logistics is the logistics sorting trolley, which is significant for bottleneck issues like excessive human dependence and poor operational efficiency in e-commerce logistics^[1]. Therefore, the research on this topic is carried out by automatically recognizing the pre-set tasks, activating the camera for image acquisition and color recognition, running the main control program in the vision recognition module, and sending the movement control command to the motion drive module to transport the sorted objects to the designated area. If the object color is not recognized, no movement command will be sent until the color of the sorted object is recognized, and this process is repeated until the object color is recognized. In order to facilitate the automation of logistics sorting, the motor module drives the logistics sorting to the appropriate storage space.

2 Overall solution design of automatic sorting trolley

The sorted objects are sorted using color recognition in this study. Red, yellow, and blue are the primary colors for the sorted objects, and red, yellow, and blue are also the three pre-determined fixed transport lanes for the three things to be sorted. When sorting items in a factory setting, the simplest sorting transport channel can significantly lower the factory operation cost^[5]. This is taking into account the actual application background of the topic. In Fig1. the schematic diagram is displayed.

* Corresponding author: 20058010@bua.edu.cn

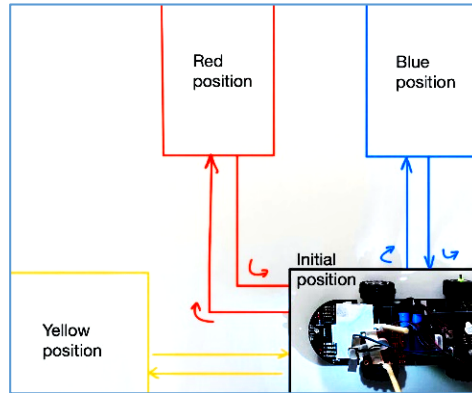


Fig. 1. Overall solution diagram.

2.1 Visual recognition module

The OpenMV3 Cam M7 model's machine vision module is used in this project to carry out the task of object classification and color recognition. The image processing module's CPU is an ARM Cortex M7 processor, which uses the module's sensors to transmit various image and video data to the microcontroller, which then processes, recognizes, and categorizes the images.

2.2 Electric drive module

The system, which is in charge of transporting the sorted objects to the prescribed placement area, uses the Arduino UNO R3 and L293D motor driver modules as the cart driving hardware. The L293D quad high-current half-H driver is used to connect the positive and negative terminals of the two DC motors in the motor drive circuit, which drives two DC motors using four PWM digital IO ports as the drive interface^[9-10]. Fig. 2 depicts the general hardware layout of the system.

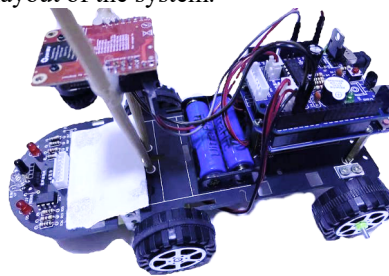


Fig. 2. Physical view of automatic sorting trolley.

3 Key technologies

3.1 Color recognition algorithm

To acquire photos of the target objects, a machine vision system based on OpenMV^[4] is developed. The captured target images are then transformed from RGB to LAB color model

space. To enhance the color segmentation of the target object, the K-means clustering technique is used in the target object image recognition and segmentation. When compared to conventional image segmentation methods for embedded system environments, this approach, which separates color blocks of the same color in the image, has the advantages of a small model and quick speed.

The lowest of these three distances is determined to be the color of the pixel point in the LAB color model by measuring the Euclidean distance between each pixel point and the average of the three colors (red, yellow, and blue)^[2]. The following equation can be used to determine how far apart the red, yellow, and blue hues are from each pixel point:

$$\Delta E = \sqrt{(\Delta L)^2 + (\Delta a)^2 + (\Delta b)^2} \quad (1)$$

where E represents the color difference from the reference color and L/a/b represents, correspondingly, the color difference between the two colors in various components. By selecting the shortest distance out of the three distances, the color of the pixel is established to be one of these colors.

The fundamental idea behind the K-means clustering segmentation algorithm, which is an unsupervised learning technique^[8], is to first randomly choose one pixel point from K pixel points as the initial cluster center, and then to compute the Euclidean distance d2 by applying the Euclidean distance to the other pixel points. The computed pixel points are grouped into the class that corresponds to the location of the closest cluster center. The clusters are then divided again until the sum of squared errors within the class is minimized, at which time the mean value of the pixel points after categorization replaces the original center^[6]. As seen in Fig 3. the image is divided into pixel points of the same color.

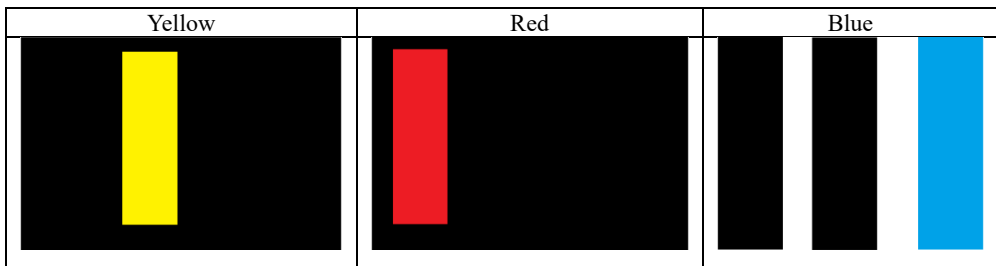


Fig. 3. Color segmentation results.

The majority of the image processing operations are included in the OpenMV vision module that was used in this project. The core algorithmic function in this project, find blobs^[3], locates and returns all of the color blocks in the image. All of the returned objects will be merged by this function. It must be called multiple times in order to recognize additional colors. Set all of the aforementioned parameters in this system to their default levels, then just add color thresholds. Such as yellow, which the blobs y object receives from `img.find_blobs([yellow threshold])`.

The find blobs function of OpenMV is used to compare the color value with the threshold during the color recognition process. If the color value is within the threshold, the color code r, b, and y are then sent to the motion driver module via the serial port. The color recognition process begins by counting the color that occupies the largest area in the current region of interest. No more color recognition will be carried out until the motion driver module idle instruction `flag=0[7]` is received in order to avoid control instruction conflict.

3.2 Serial communication design

Through a serial UART and a TTL level connection, the visual recognition module OpenMV and the motion driver module Arduino UNO R3 communicate. Set the baud rate to 19200 and activate UART1 in the OpenMV configuration for the visual recognition module. Enable softSerial1 and set the baud rate to 19200 in the motion driver module settings.

When a red paper jam is detected, the vision recognition module sends the character "r" over the serial port; when a yellow paper jam is detected, the character "y" is sent through the serial port; and when a blue paper jam is detected, the character "b" is sent through the serial port.

3.3 Motor drive circuit

The Arduino's analog IO port is used to output the motor's control signal as a PWM pulse-width modulated signal. L239D then amplifies the output signal to increase the current, which is then supplied to the DC motor's internal circuit to drive motor rotation. To meet the goal of offering low speed and great torque, a matching gearbox is additionally provided.

Since the amount of friction that a car's tires have with the ground changes depending on the scene, this system's motor drive design adjusts the PWM pulses' parameters using the enumeration method, and then sets the PWM pulse value of the control signal for the two motors to 130 (the Arduino analog port's PWM output range is 0–255).

When serial information is received in the serial port cache, the data is removed for command parsing, and the path is moved in accordance with the corresponding color code. The motion driver module serial port uses a soft serial port that is simulated by the IO port, and the baud rate is set to 19200.

4 Conclusion

The automatic sorting trolley built on OpenMV is designed using image processing, motor control, and serial communication technologies. A vision recognition module and a motion drive module make up the system. Serial communication is employed to communicate the target color information and regulate the motor, and the LAB color space model is used to realize the recognition of the color scale of goods.

The research work of this paper supported by the Beijing scientific research program: Research on Intelligent identification system Of peach pests based on Internet Of things(km201810020018) .

References

1. X.Y.Liu, Adhesion, **47**, 109-112 (2021)
2. H.J.Zhang, Fujian J, **27**, 146-147 (2011)
3. X.H.Wang, Zhejiang Sci-Tech University, **02**, 54 (2020)
4. N.N.Huang, Acta Agric.Univ.Jiangxi, **07**, 61 (2020)
5. D.Wang, *Manufacturing Automation(WJST)*, **43**, 6-9+20 (2021)
6. K.Y.Chen, C.Y.Wu, Z.H.Guan, H.T.Li, G.Wang, Jiangsu J, **37**, 1425-1435 (2021)
7. S.J.Xia, H.Yang, W.Q.Ai, Journal of Changshu Institute of Technology, **35**, 59-64 (2021)

8. Q.R.Ge, Xidian University, **05**, 59 (2012)
9. T.G.Jiang, Shenyang University of Technology, **02**, 81 (2015)
10. Y.W.Tian, R.X.Lv, Y.J.Li, China Science and Technology Information, **02**, 81-83 (2020)