Design of Intelligent Vehicle Path Recognition System Based on LabVIEW Image Processing

Lijun Wang[1*], Yuan Liu[1], Yahui Liu[1], Qingsheng Hu [1], Jingzheng Hu[1]
[1]North China University of Water Resources and Electric Power, Zhengzhou, China
*Corresponding author: Lijun Wang
E-mail:_wljmb@163.com, mrliu89@qq.com, 1016005115@qq.com, 1415675421@qq.com, 649393155@qq.com_

Abstract

The design boldly uses the image processing algorithms on LabVIEW and NI myRIO to control the intelligent vehicle. The software design includes the control of servo, motor and camera, NI myRIO calls and image processing. The processing includes the image binarization, filtering, edge detection, establishing the coordinates, angle measurement image matching and image color recognition processing. NI myRIO does some FPGA processing on the detected angle, and it gives some feedbacks for the angle of steering to the rotation gear in real time, so as to realize the automatic tracking and automatic warehousing of the intelligent vehicle.NI myRIO realizes control of the intelligent vehicle for motor: the speed, stop and the rotation of the motor by identifying the detected image and color. The hardware design includes the motor driving circuit and the buzzer driving circuit parts.

Keywords: LabVIEW, intelligent vehicle, NI myRIO, image processing.

1. Introduction

The design of intelligent vehicle path recognition system based on the LabVIEW image processing uses NI myRIO equipment and WiFi communication technology. A PC software of intelligent vehicle control system based on NI myRIO call and image processing algorithm is designed and developed by the Vision of LabVIEW software, and the intelligent vehicle path recognition procedures are built to realize the path recognition of the intelligent vehicle.

2. The scheme of overall design

Path recognition system of intelligent vehicle based on NI myRIO is the control core of the system. It is equipped with CMOS camera sensor, with the basic power circuit and motor drive circuit, which can realize the acquisition, storage, transfer, binarization, filter, edge detection, establishing coordinate, angle detection of image by LabVIEW language[1, 2] and call the camera to capture images in the LabVIEW programming. And then the sensor transforms angle to servo motor rotation through a number of algorithms, so as to realize the automatic tracking and warehousing of intelligent vehicle and so on. By setting the image and color detection, the control of motor rotation speed and rotation by NI myRIO is realized, and the smart car stops when meeting the red signs, decelerates when meeting the green signs and successfully turns when meeting other colors are simulated. By calling the Z axis parameters of NI myRIO's three axis accelerometer, the intelligent vehicle can move at different velocities in different road and traffic conditions. The monitoring system is controlled by the host computer monitoring software

LabVIEW, which is loaded on the PC, so as to receive, display and send the speed, corner, road condition information and other data[3] of the intelligent vehicle, so as to monitor the movement of the intelligent vehicle in real time. The buzzer can achieve intelligent vehicle's alarm function when the steering angle goes to a certain angle and bad road conditions appear. The realization of these functions is based on the feedback angle of intelligent vehicle image processing. So, the core of the entire system is the camera. Fig. 1 shows the overall design scheme of intelligent vehicle control system.
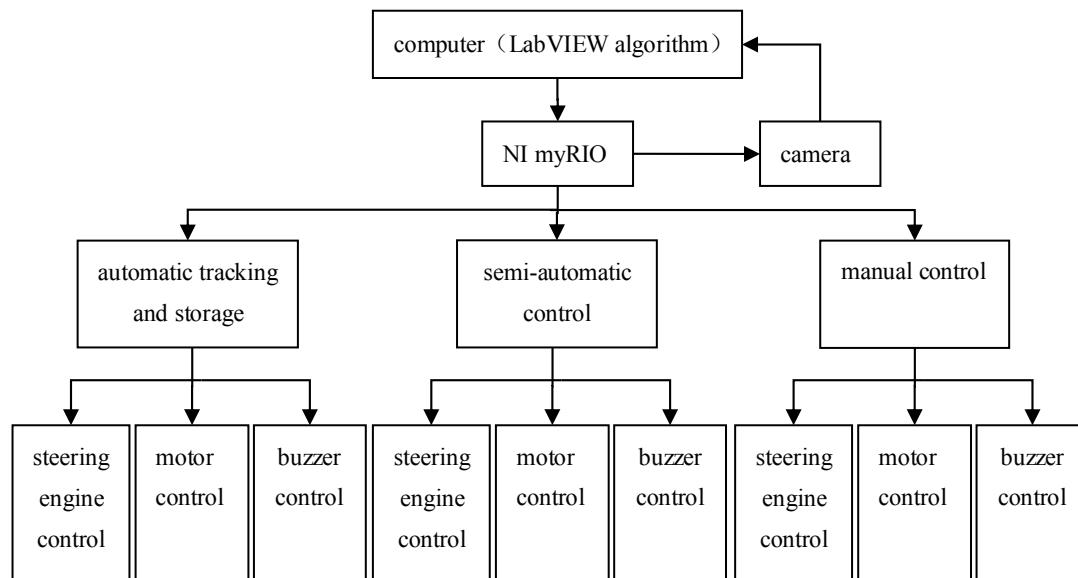

Fig. 1. Overall design scheme of intelligent vehicle control system.

3. Hardware components

Original intelligent vehicle control system will choose a large number of sensors, and the intelligent motion of intelligent vehicle is realized through the sensor data transmission, and the intelligent vehicle based on LabVIEW control system has only one camera sensor[4]. Information obtained by selected camera is directly related to the stability of intelligent vehicle control system and a variety of unexpected situations, so we must choose a camera of high performance-price ratio, good stability and high real-time processing performance.

Compared to the certain limitations of other sensors for automation designers, the high-definition camera with USB plug can be easily applied to a variety of car plug. Thus, the working voltage of the camera sensor is +5V, so the vehicle slot and the computer can call the image, which is convenient.

Through the WIFI signals, the host computer controls NI myRIO directly on the LabVIEW platform, then NI myRIO controls camera, motor, servo and buzzer, then the motor and servo control the movement of intelligent vehicle directly. NI MyRIO has three control channels A, B and C, and each channel has a number of different PWM ports and DIO ports, which can achieve the control by inserting the steering engine, motor, buzzer, etc. into NI myRIO. NI myRIO comes with the USB slot, and the camera with USB plug can be inserted to achieve NI myRIO image acquisition and subsequent image processing[5]. The control principle of NI myRIO is shown in Fig. 2.
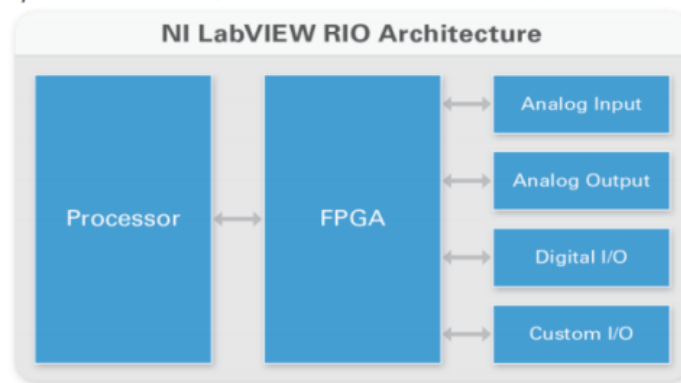
Fig. 2. Control principle of myRIO NI.

NI myRIO includes six analog outputs, 10 analog inputs, up to 40 digital I / O lines and audio I / O channels, a triaxial accelerometer, a on-board WiFi and a plurality of programmable led, which are integrated in a durable, closed architecture[6]. Its pin diagram is shown in Fig. 3.
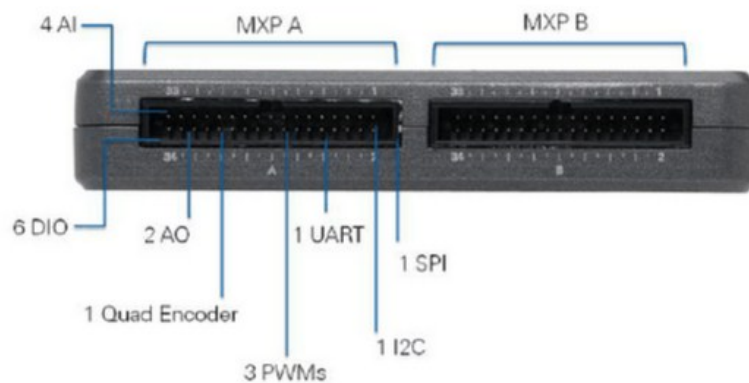

Fig. 3. Control pin diagram of NI myRIO.

The control of steering engine, motor and buzzer is realized through the control of Ni myRIO PWM export and voltage pin. The pins and camera are controlled by a USB slot and voltage, at the same time, the Z axis acceleration of intelligent vehicle under bad road situation is measured using the built-in Ni myRIO on-board acceleration sensor.

4. Software components

The hardware settings consist of a steering gear, an electric motor, a buzzer and a camera serial port configuration, a steering gear and a motor PWM setting, a camera opening and an image transmission system. Algorithm processing is mainly the image processing algorithms, including the binarization, filtering, edge detection, the establishment of coordinate and angle measurement. Algorithm processing also includes control algorithm of the steering engine and motor speed, and the control algorithm of buzzer call. The software overall trend is shown in Fig. 4:
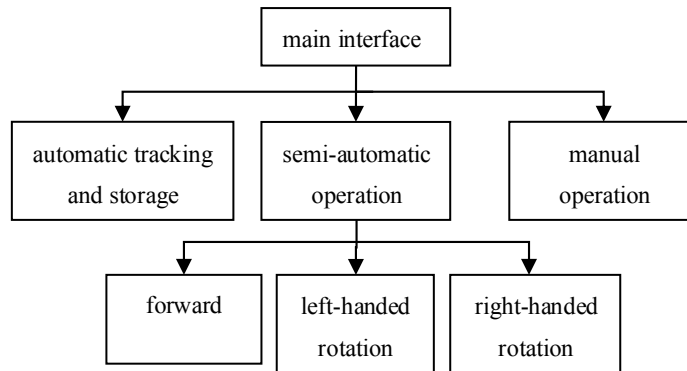
```
                              ┌──────────────┐
                              │main interface│
                              └──────────────┘
          ┌──────────────────────┼──────────────────────┐
          ▼                      ▼                      ▼
  ┌────────────────┐    ┌────────────────┐    ┌────────────────┐
  │automatic tracking│   │ semi-automatic │    │    manual      │
  │  and storage    │    │   operation    │    │   operation    │
  └────────────────┘    └────────────────┘    └────────────────┘
              ┌──────────────┼──────────────┐
              ▼              ▼              ▼
        ┌──────────┐  ┌──────────┐  ┌───────────┐
        │ forward  │  │left-handed│  │right-handed│
        │          │  │ rotation │  │ rotation  │
        └──────────┘  └──────────┘  └───────────┘
```

Fig. 4. Overall trend of system setting.

Although the design is path recognition of intelligent vehicle based on the LabVIEW image processing, it still needs to configure the hardware. Presented here is for hardware function of LabVIEW, so the main software flow of the system as shown in Fig. 5 is adopted when controlling the hardware:
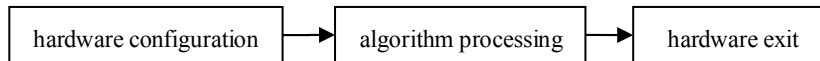
```
┌────────────────────────┐    ┌──────────────────────┐    ┌──────────────┐
│ hardware configuration │ ─► │ algorithm processing │ ─► │ hardware exit│
└────────────────────────┘    └──────────────────────┘    └──────────────┘
```

Fig. 5. Main software flow of the system.

In order to let users clearly understand the movement velocity and the surrounding environment of intelligent vehicle and to facilitate debugging, the image monitoring and speed panel display widgets are added in the front panel of the program in tracking forward and automatic storage[7]. The front panel of the program is shown in Fig. 6.
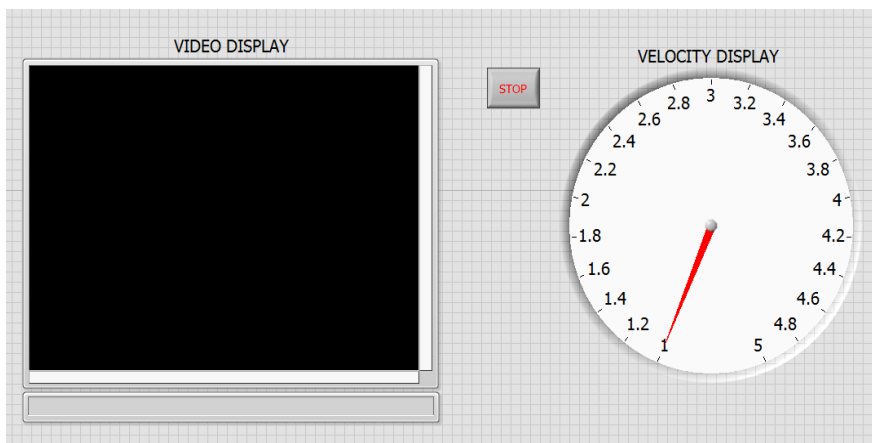

Fig. 6. Automatic tracking and storage of the front panel.

4.1. Hardware configuration of NI myRIO

The communication between NI myRIO and host computer can be realized by configuring the I/O port of NI myRIO. And the I/O port of the motor and steering engine is directly configured with PWM, so that the PWM signal has a communication connection with NI myRIO I/O port[8]. At the same time, the camera's I/O port is directly configured with image connection and acquisition connection. When using

the LabVIEW language to control the hardware, it's necessary to be familiar with the LabVIEW language programming environment, instead of writing all languages in a WHILE loop and letting the program execute by itself. The hardware must be configured in the outside of big WHILE loop, otherwise in every execution it's necessary to reconfigure the hardware, which makes the loop complex and makes NI myRIO running messy easily. The rear panel of NI myRIO's all hardware configuration is shown in Fig. 7.
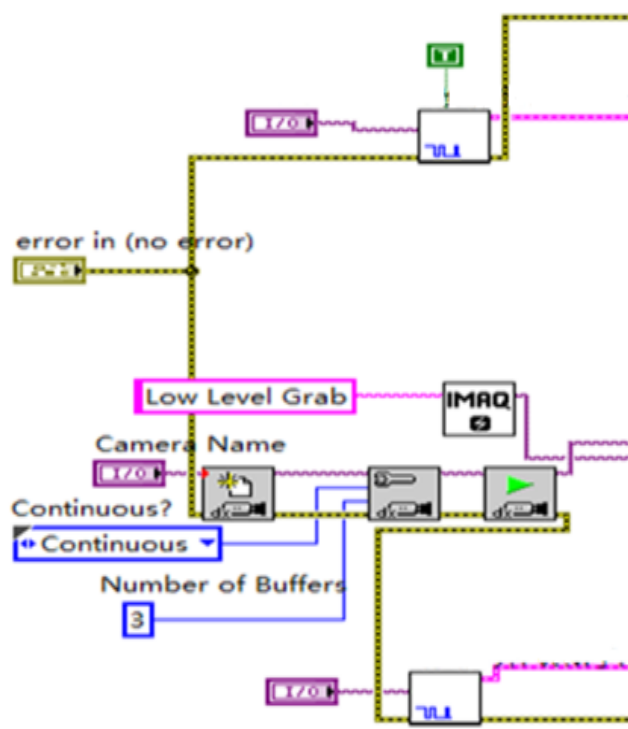

Fig. 7. The rear panel of the hardware configuration.

Here, the final PWM opening sub-VI is called, and this function sets the NI myRIO transfer and the input of NI myRIO PWM signal. If the PWM port settings are not set properly, it will cause the PWM signal invalid. And the development function of FPGA call should be used to run the program, or the data may show that the configuration is low.

4.2. NI myRIO control algorithm

The image processing algorithm is mainly reflected in the automatic tracking and storage algorithm[9]. Algorithm can achieve the intended purpose by the followed image processing sub-VIs: image cache < image color transfer < image binarization processing < image filtering and image edge detection < image edge processing < establish of image coordinates < image angle measurement. After adding the image color recognition, the rotation of the steering engine angle and the rotation speed of the motor can be achieved by examining the color of the image on the road.

In storage operation, steering engine executes according to the algorithm so as to realize the automatic storage of the intelligent vehicle. If not, it enters into the automatic tracking procedure. In the automatic tracking procedure, image is binarized at first, which is for the next step of image edge detection and edge processing. The next step is to establish coordinates and angle measurement. Through a series of image processing, finally the deviation of the angle of intelligent vehicle can be measured, and it is fed back to

the intelligent vehicle steering engine in real time. Then intelligent vehicle steering engine transforms the measured angle into the angle that the steering engine rotation is needed, to realize the automatic tracking of intelligent vehicle by some specific algorithm. And buzzer algorithm is also reflected in this part. If the both-side corner of the steering engine exceeds 10 degrees, the buzzer will ring. If the identified color is red, the motor will emit a tweet, which implies a danger. If the pavement condition is not good and the value exceeds the set value of three axis acceleration sensor, the buzzer will also ring. At the same time, the motor will make the response to reduce the speed.

Each image processing sub-VI is chosen in order to simplify the function and the page in the processing function of image processing algorithm. In the process of image processing, the sequence of selected image processing is as follows: image extraction sub-VI < edge detection sub-VI < edge processing sub-VI < establish the coordinates sub-VI < angle measurement sub-VI. The following is a description of these sub-VIs respectively.

4.2.1. Image extraction sub-VI

After entering the WHILE cycle, the image starts to cache. After cache, the pigment of image is extracted, then whether the settings for the parameters are accurate or not is need to notice. If the Color Plane settings reduce any one color, the image will be distorted before the start of processing, and there is no significance to perform a variety of processing. An image extraction sub-VI is shown in Fig. 8.
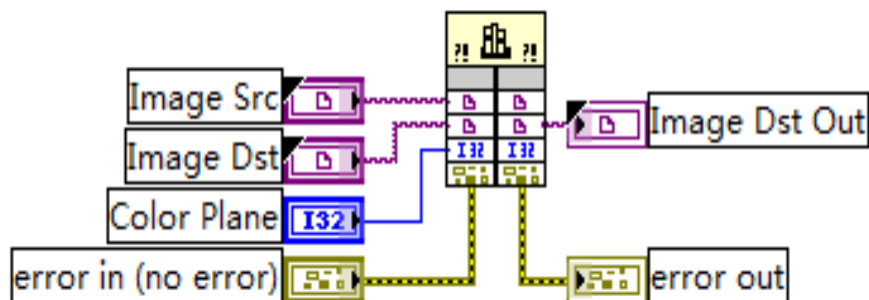

Fig. 8. Image extraction sub-VI.

4.2.2. Edge detection sub-VI

When compiling the edge detection sub-VI, the setting of the image ROI needs to be noticed, and the ROI area can't be too large or too small. A too large area will interfere with the image processing and a too small area makes the edge can't be detected to realize the expected processing. After setting the ROI area, the direction of the detected straight edge should be set. If not, a good choice about reference point of coordinate and angle measurement is improbable. So the first step to deal with the image is to set all the parameters properly. When setting these parameters, an image display widget needs to be set, and to set parameters once again according to the image display widget until the appropriate parameters are found. Because the image recognition can be carried out only by detecting a straight edge in this function, it's not necessary to set the coordination and selection of a number of sides. The edge detection sub-VI is shown in Fig. 9.
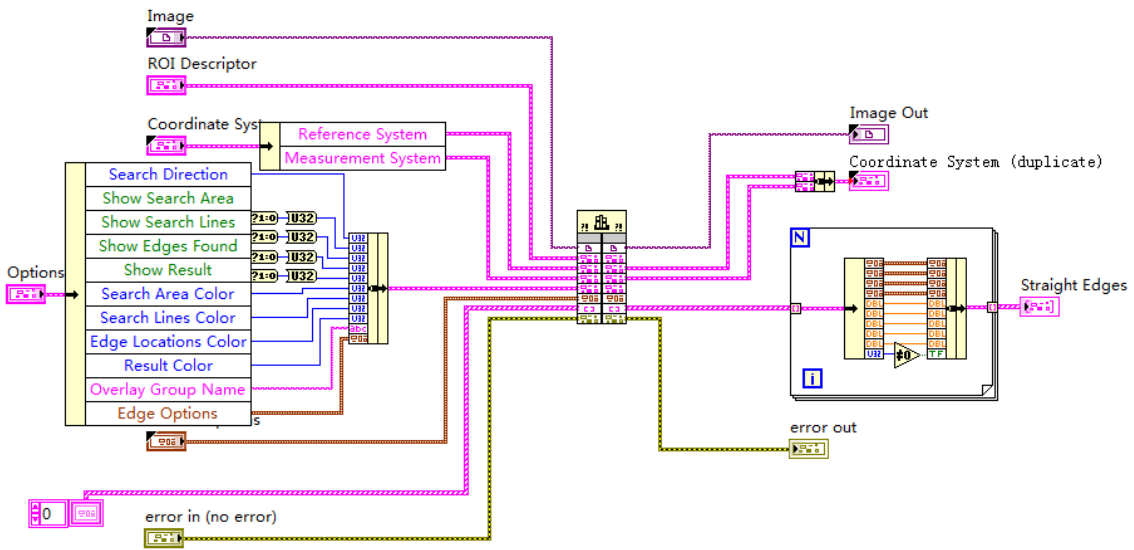
Fig. 9. Edge detection sub-VI.

4.2.3. Edge processing sub VI

After edge detection, the edge processing needs to be carried out. In the edge processing, the main function is to set the location and order of the coordinates and points for detected straight edge. When debugging, it's also necessary to set an image display widget in the function, which is shown on the front panel by image display, until to find the right parameters. The edge processing sub-VI is shown in Fig. 10.
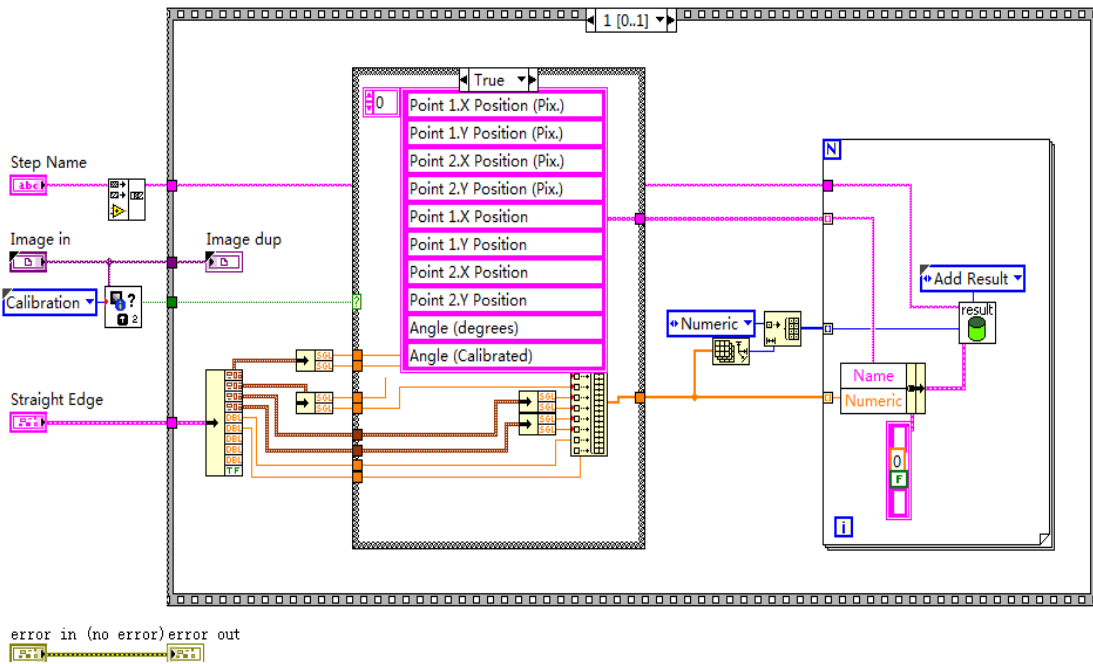

Fig. 10. Edge processing sub-VI.

4.2.4. Establish the coordinates sub-VI

The angle of straight edge can't be directly measured after straight edge's detection, so a reference system, that is, the coordinates of the system needs to be set up. In this way, the angle difference value of the straight edge can be specifically detected. There is a question in setting the coordinate system. That is, when the intelligent vehicle is going forward, the first detected point is the nearest place from the intelligent vehicle, which is the needful coordinate set reference point. So the coordinate system is built up on the nearest point from the intelligent vehicle. Fig. 11 shows the sub-VI of the coordinates establishment.
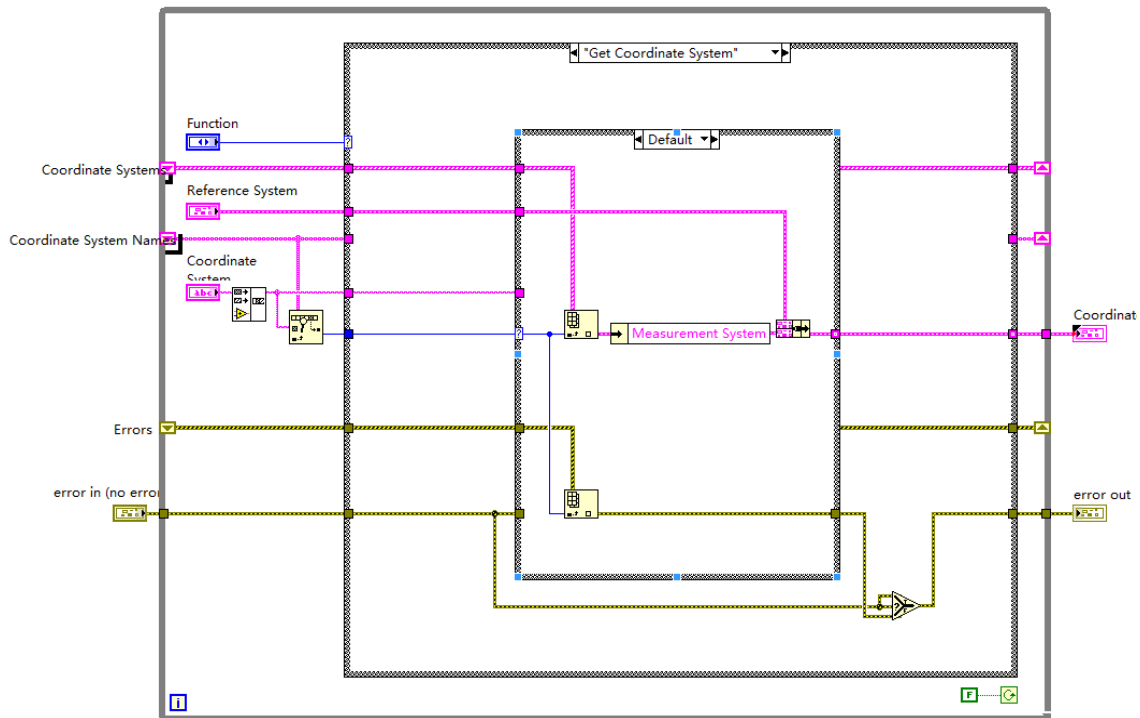

Fig. 11. Establish the coordinates sub-VI.

4.2.5. Angle measurement sub-VI

In the angle measurement function, the reference coordinates and the origin needs to be chosen. Here the point 2 is chosen as the origin and the Y-axis as the reference. When setting up the coordinate system, how to establish to simplify the subsequent angle transition of the steering engine needs to be considered. Certainly, the debugging process not only needs to carry out step by step at the same time looking at the image display, but also needs to transfer the output original data after the angle measurement process and the steer angle respectively, and the reference axis and the origin ultimately are determined by comparing and analysis. Fig. 12 is the angle measurement sub-VI.
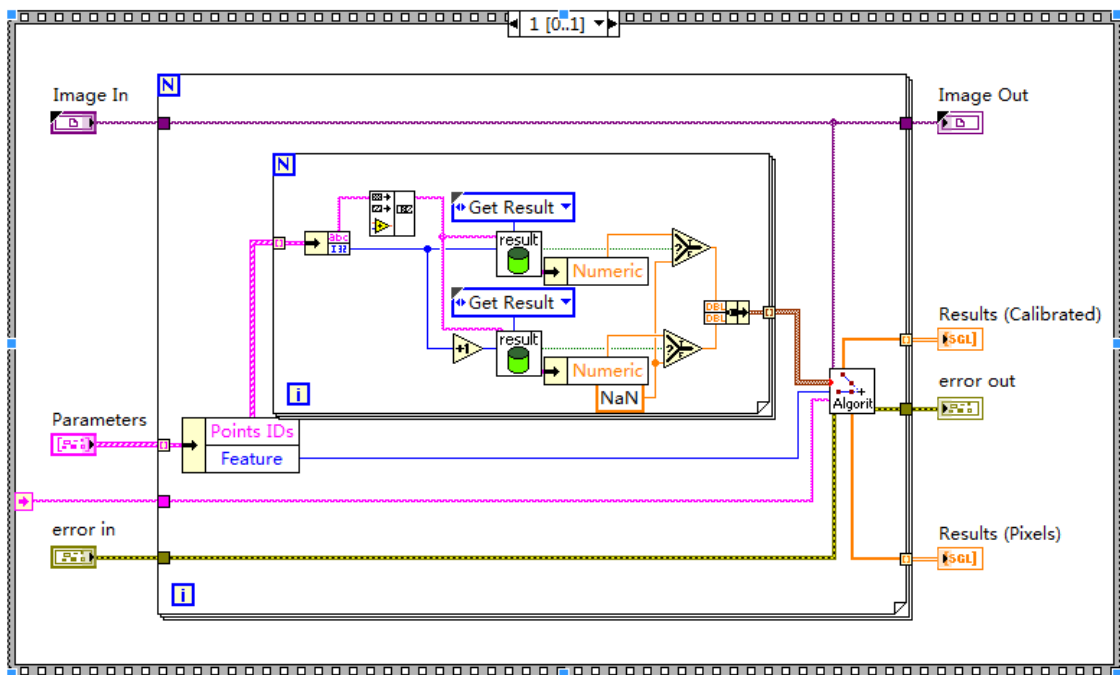
Fig. 12. Angle measurement sub-VI.

The algorithm processing is executed in a certain order, but it doesn't use sequential function. The sequential function will cause errors of the function structure data flow, so it directly uses the error cluster instead of sequential function. Then not only the redundancy of function decreases, but also the error can be timely discovered and checked (in the case of highlighted). Automatic tracking and storage algorithm conditions are shown in Fig. 13.
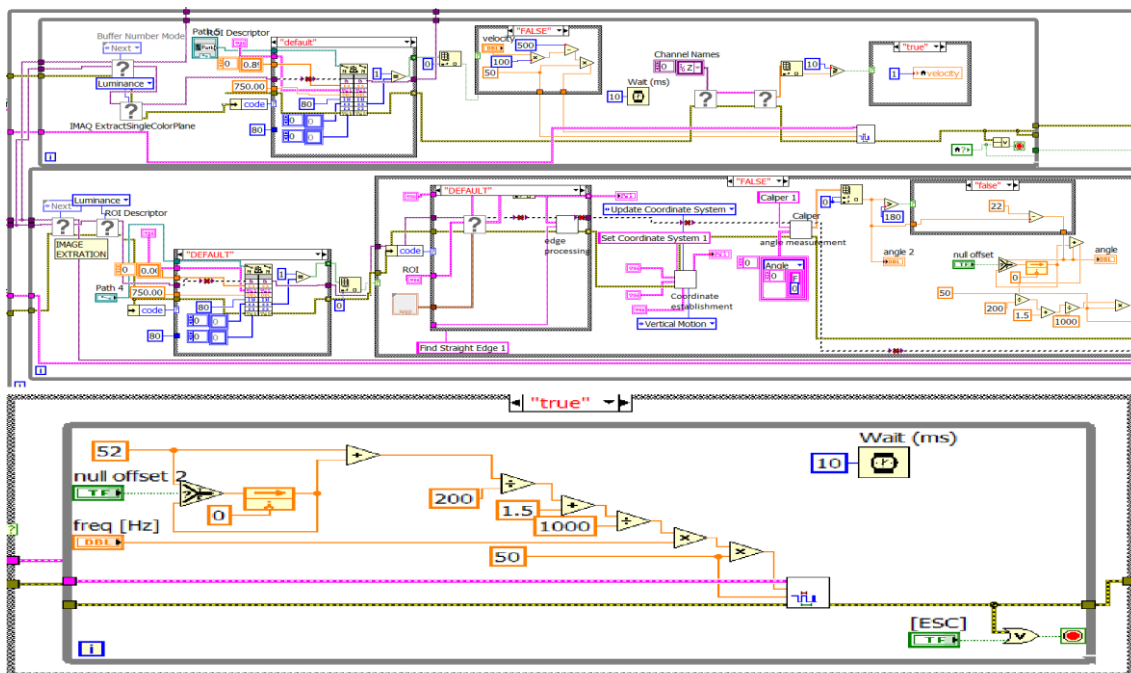

Fig. 13. Automatic tracking and storage algorithm conditions.

4.3. NI myRIO hardware exit

When compiling the hardware exit program, according to the direction of data flow, firstly NI myRIO stops the camera dialogue orderly. Secondly it erases the camera image and exits the camera. Finally it releases the image. When compiling the front of the algorithm processing, there are several WHILE loops need to be used. In order to avoid multiple Boolean stopping to make the program confusion, local variables are selected to control a number of WHILE cycle's stop[10]. But the button cannot bounce up, it means that it is time to start again after each time the stop button is pressed. The Boolean control module is added to the hardware exit section for this problem, it is shown in Fig. 14.
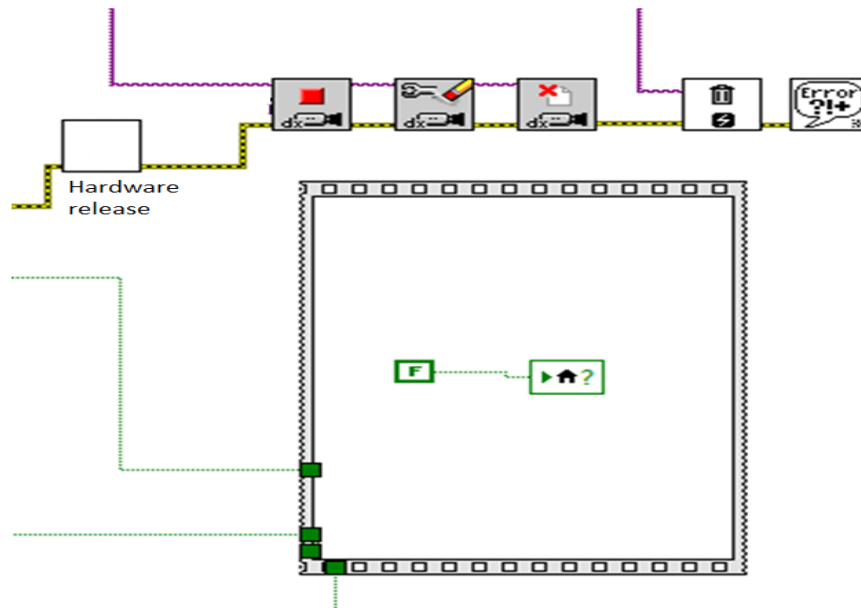

Fig. 14. Hardware exit and Boolean control module.

5. Conclusions

The design of intelligent vehicle path recognition based on LabVIEW image processing lies in that it can display the speed and the road condition information in real time, and the road condition analysis based on the image processing algorithm can make the system more intelligent and stable. Such a system not only is convenient for real-time monitoring of the system, but also makes the intelligent vehicle more comprehensive, intelligent and adaptable. The intelligent vehicle real-time monitoring test image is shown in Fig. 15.
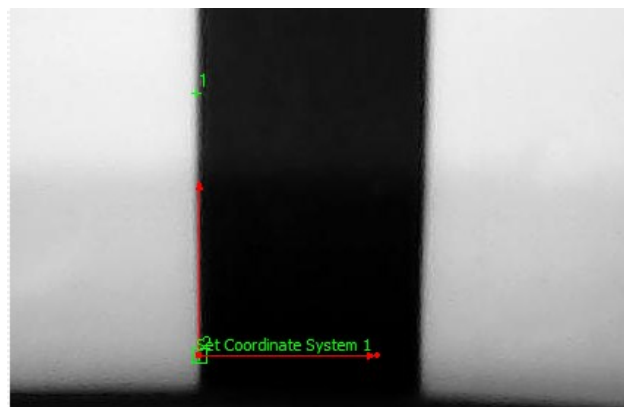

Fig. 15. Intelligent vehicle real-time monitoring test image.

Acknowledgements

References

[1]   B Wang, K Zhang, M Liang, W Luo. "Non-Contact Vision Inspection System of Geometric Dimension Based on LabVIEW and IMAQ Vision." *Advanced Materials Research* 305 (2011): 53-56.

[2]   Li Pei-Jiang, Li Shu-Qing, H Gao, T You, L Meilei. "Network machine vision inspection system based on LabVIEW." *Transducer & Microsystem Technologies* 26.8 (2007): 26-28.

[3]   Wang Li-jun, Liu Yue, Huang Yong-liang. "Virtual Signal Generator and Oscilloscope Design Based on LabVIEW." *Journal of North China Institute of Water Conservancy and Hydroelectric Power* 31.3 (2010): 57-59.

[4]   Li Wei, Ren Na-na, Hu Yan-xia, Ren Qing-qing, Zhu Shao-jun. "The design of automatic testing system for pointer type instrument based on machine vision and LabVIEW." *Automation & Instrumentation* 186 (2015): 178-181.

[5]   Zheng Shu-bin, Chai Xiao-dong, Han Guo-ge. "Track Gauge Inspection Method Based on Computer Vision." *Urban Mass Transit* 13.9 (2010): 73-76.

[6]   Blume P. A. *The LabVIEW Style Book (National Instruments Virtual Instrumentation Series)*. Prentice Hall PTR, 2007.

[7]   Chouder Aissa, Silvestre S, Taghezouit B. "Monitoring, modelling and simulation of PV systems using LabVIEW." *Solar Energy* 91 (2013): 337-349.

[8]   Madisetti Vijay. *Digital signal processing fundamentals*. CRC press, 2009.

[9]   Sui Xiang, Wang Qi, XUE Peng-jun. "Development of smart car comprehensive information monitoring system based on LabView." *Modern Electronics Technique* 35.20 (2012): 149-151.

[10] Du Xiang-jun, Miao Chang-yun. "A Designing Virtual Instrument Based on EPP." *Control & Automation*, 315 (2005): 129-130.