[1]Zijie Lu

Recognition and Picking of Target Fruits of Crop Picking Vehicles based on Computer Vision

Journal of Electrical Systems

Abstract: - A crop-picking vehicle was designed to automatically pick crop fruits and reduce human labor in the later harvest. The intelligent vehicle uses a Raspberry Pi 3B + controller, combines with a monocular camera, infrared sensor, four-degree-of-freedom manipulator and manipulator, and adopts the open source computer vision library OpenCV to realize the recognition and extraction of the target fruit. Based on the actual test verification, it was concluded that the intelligent vehicle could successfully realize the picking test of simulation citrus under the condition of good ambient light.

Keywords:Smart picking vehicle, Computer vision, Robotic arm, Raspberry Pi, OpenCV

I. Introduction

Agricultural robot is the frontier field of international agricultural mechanization research. Currently, there are also studies on picking robots in China, but compared with industrial robots, picking robots have more complex structures and control systems, higher production costs, and higher technical levels and costs for equipment use and maintenance[1]. Therefore, in this paper, an agricultural smart vehicle based on Raspberry Pi 3B+controller (hereinafter referred to as Raspberry Pi controller) was designed..

II. Hardware design

The hardware of asmart picking vehicle consists of four parts: a mobile platform, an electronic control module, a sensor, and a mechanical arm device. The mobile platform includes a DC motor, a battery power supply, and a tracked chassis; The electronic control module includes a Raspberry Pi controller, a DC motor drive module, a steering gear control module, and a step-down module; The sensor includes a monocular camera and an infrared ranging sensor; The mechanical arm device consists of two parts: the mechanical arm and the manipulator. The mechanical arm and the manipulator are composed of a mechanical support and a steering gear. The structure diagram of the smart picking vehicle is shown in Figure 1.

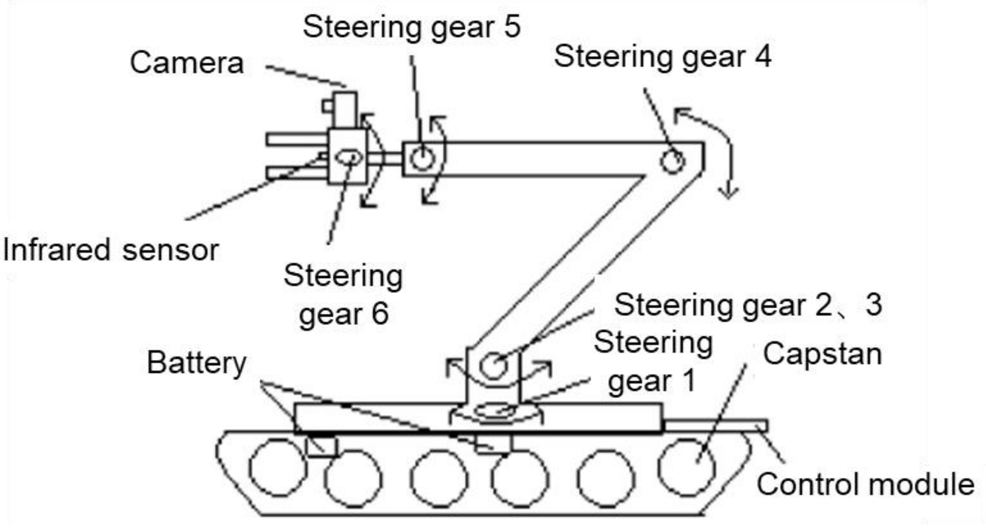

Figure 1 Solid structure diagram of smart picking vehicle

A smart picking vehicle uses a total of 5 steering gearcontrol mechanical arms and 1 steering gear control manipulator. Among them, steering gears 2 and 3 jointly control a joint, which can ensure that the mechanical arm is extended for a long distance to avoid excessive torque and loss of control or damage. The camera and infrared sensor are fixed at the end of the mechanical arm and change with the change of the end position of the

[1]College of Computer and Information Engineering, Nanjing Tech University, Nanjing, 211816,China

mechanical arm. The control module is all fixed on the plate at the back end of the smart vehicle, avoiding the working range of the mechanical arm. The capstan is linked to the DC motor to drive the track to rotate.

The hardware composition of each part of the smart picking vehicle is shown in Figure 3, and the arrow direction represents the flow direction of the sensing signal or the control signal.

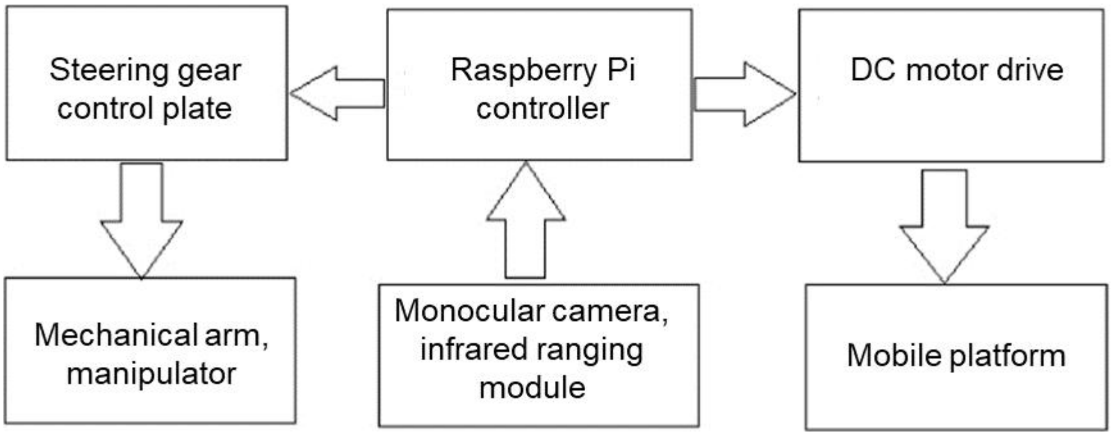
Steering gear control plate
Raspberry Pi controller
DC motor drive
Mechanical arm, manipulator
Monocular camera, infrared ranging module
Mobile platform

Figure 2 Hardware composition relationship of smart picking vehicle

A. Mobile platform

A crawler chassis is selected instead of an ordinary tire chassisto ensure that the smart picking vehicle has better mobility in a muddy environment. The left and right tracks of the mobile platform have one driver and five load-bearing wheels, and the left and right drivers are driven by a DC motor, respectively. The DC motor parameters are as follows: the rated operating voltage is 9V, the maximum load current is 1200mA, the load torque is 3000g·cm, and the output rate is 150r/min. In actual work, the input voltage of the DC motor is 7.2V, which does not reach the rated voltage but has met the mobile demand.

B. Electronic control module

The main controller is an ARM-based Raspberry Pi 3B+microcomputer mainboard. Its size is only that of a bank card. Its central processor is a 1.4GHz 64-bit 4-core ARM Cortex-A53 CPU. It has 4 USB 2.0 interfaces, 1 HDMI video output interface, 1 Gigabit Ethernet interface, 1 wireless network card, 40 peripheral pins, including 28 GPIOs, which can realize serial port communication,I2C communication, SPI communication, and digital level input and output. Others are used as power pins and ground pins[2]. Also, it is equipped with a SD card with a capacity of 16GB. As there are four USB interfaces, the Raspberry Pi controller can be connected to the USB mouse, keyboard and USB camera to meet the needs of video acquisition and programming. The main advantages of using Raspberry Pi controller include low price, low power consumption, and multiple functions, while the disadvantage is that it is difficult to meet the requirements for complex image processing operations.

C. Sensor

The sensor consists of two parts: a monocular camera and an infrared ranging sensor. The monocular camera is a 720 P wide-angle camera of Linbaishi's UVC driverless protocol. It uses a USB 2.0 interface with a frame rate of 30 FPS. The image storage format defaults to JPG format and can be used directly on the Raspberry Pi controller. In practical application, the resolution is set to 480P, which can already meet the use conditions.

The infrared sensor has a transmitting tube and a receiving tube. The signal output end can return high or low levels according to whether the receiving tube receives the reflected infrared signal. The adjustable resistance of the infrared sensor can be adjusted to detect objects in an appropriate range.

D. Mechanical arm device

The mechanical arm is a 6-degree-of-freedom manipulator. In fact, only 4 degrees of freedom are used to complete the rotation and movement. It can reach any three-dimensional coordinate point within the grasping range, which meets the control requirements and relatively reduces the control difficulty. The maximum grasping radius of the mechanical arm is 40 cm. The manipulator is installed at the end of the mechanical arm, with a pair of symmetrical side-by-side fingers, which can only complete the opening and closing action. The degree of opening is controlled by a steering gear, and the direction of opening and closing motion is parallel to the horizontal plane.

The steering gear model that controls the rotation of the mechanical arm is HM-MS10.The rated voltage is 7.2V, the stop torque is 20kg·cm, and the rotation limit angle is 180°. The driving method of the 180° steering gear is: the input cycle at the signal end is a level of 20 ms, in which the high level time changes from 0.5ms to 2.5ms, corresponding to the steering gear rotation angle of 0° to 180°[3].

III. Program design of smart picking vehicle

The program of the smart picking vehicle system mainly consists of the following links: image acquisition, target recognition, target positioning, vehicle movement, and targetgrasping. The 20170907 version of Raspbian-stretch system loaded in the SD card of the Raspberry Pi controller is a Linux-based free and open-source operating system[2].The system has a built-in Python3.5 interpreter, which can cooperate with other Python libraries to realize serial communication, GPIO and camera operation. The program design process is shown in Figure 4.

A. Image acquisition

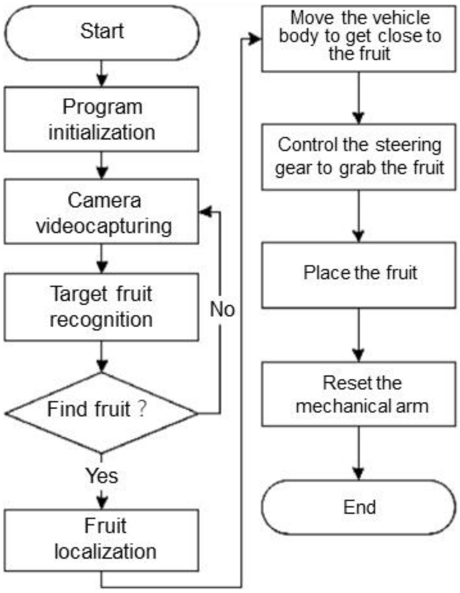

Figure 4 Program design process

Image acquisition and subsequent image processing steps aim to process images. OpenCV 3.1.0 is used here. OpenCV (Open Source Computer Vision Library) is a cross-platform computer vision librarybased on a BSD license (open source) and can run on Linux, Windows, Android and Mac OS operating systems. It is lightweight and efficient and is composed of a series of C functions and a small number of C + + classes. Meanwhile, it provides the interface of Python, Ruby, MATLAB and other languages and realizes many general algorithms in image processing and computer vision[4]. Using OpenCV can reduce the concern about the internal complex principles of image processing. In general, the coding efficiency is more perfect and efficient than that written by individuals, and it is easy to debug and implement complex image processing.

It is necessary to download and compile OpenCV3.1.0 in the Raspbian-stretch operating system to generate a library that can be called by Python3.5so that OpenCV functions can be used to implement image processing[5]. By invoking the VideoCapture class to instantiate a video object, and then using the read method of the class, the state of whether the camera video is successfully read and the image captured by the camera can be obtained[5]. The size of the captured image is 320 pixels wide and 240 pixels high.

B. Target recognition

1)Color space conversion

The image information read directly from the camera is in the form of BGR three primary colors, so it is necessary to convert the color space from BGR to HSV to obtain better discrimination. H represents hue, S represents saturation and V represents brightness. In OpenCV, using the cvtColor function and passing the parameter COLOR _ BGR2HSV, the color space conversion of the source image can be realized, and the image of the HSV color gamut can be obtained [5].

2) Determine the color threshold and binarization

After the image is converted from the BGR color space to the HSV color space, the thresholds of H, S, and V are set, respectively, and the binarization operation is performedto distinguish the part of the image within the threshold range from other parts and highlight the target area to be found. Using OpenCV's inRange function, it is possible to set the part within the image threshold range to white and the other part to black. The inRange function requires the upper and lower thresholds of the three HSV channels. In Python, the upper and lower thresholds of the three channels are passed into the inRange function with tuple-type parameters, and the returned image is the binarized image[6].

C. Target positioning

The specific idea of target recognition is as follows: after the binary image is obtained, the contour information is found by using the method of finding the target contour; then, the number of contour pixels is calculated, and the appropriate threshold of pixel number is selected to remove the contour part with a small area. Later, the circumscribed rectangle of the filtered contour is found, and the position information of the circumscribed rectangle, that is, the position of the target in the image, is obtained.

1) Extraction of contour

In OpenCV, the findContours function can be used to extract the contour of the binary image[5]. The input image of the function must be binary, and there are two incoming parameters to be selected. The RETR _ EXTERNAL parameter is used here, indicating that only the outermost contour is extracted, and the CHAIN _ APPROX _ SIMPLE parameter represents the compressed horizontal, vertical and diagonal line segments, retaining only their endpoints. The function returns the number of contours and the list of contour information, and the pixel information of each contour can be obtained by traversing the list of contour information.

2) Screening contour

The basis for screening the contour is the area of the contour, that is, the number of pixels surrounded by the contour. Using the contourArea function in OpenCV, the incoming parameter is the contour information returned by the findContours function, and the return value is the surrounding area of the contour[5]. After obtaining the area surrounded by each contour, a threshold is selected to remove the contour that is less than the area threshold, aiming to prevent misjudgment of the target caused by other interference. After testing, the effect is better when the threshold is 8000.

3) Get the contour-circumscribed rectangle

In the computer image, a pixel point determines the coordinates through the two-dimensional Cartesian coordinate system. Usually, the first pixel point in the upper left corner is the origin, that is, the first point coordinate in the upper left corner is (0, 0), the right direction is the positive direction of the x-axis, and the downward direction is the positive direction of the y-axis. OpenCV also uses this coordinate representation.

The contour-circumscribed rectangle can be obtained by the boundingRect function in OpenCV. The incoming parameters are the contour information after area screening. The function returns the upper left corner coordinates (x, y) of the contour-circumscribed rectangle and the width and height of the rectangle (w, h) [5], and then the center point can be calculated, namely (x+w/2, y+h/2). So far, the location information and size information of the target are obtained.

D. Vehicle body moving

To make the target enter the range of the manipulator, the vehicle body needs to move to the target if necessary. When the target is found, the distance between the citrus and the manipulator is roughly judged according to the size of the circumscribed rectangle of the target. The basis for judging whether the citrus is within the grasp range of the manipulator is the area of the circumscribed rectangle. If the area of the circumscribed rectangle exceeds the threshold, it is judged to be within the grasp range of the manipulator, otherwise, it is judged not to be within the grasp range. At this time, the vehicle body needs to approach the target. The vehicle body steering can be realized by differential control of the motor.When the target'scircumscribed rectangle is in the central area of the image, it is judged that the steering is completed, and then the vehicle body moves forward. The vehicle body corrects the steering deviation during continuous forward and steering until the circumscribed rectangular area of the target exceeds the threshold, and the grasping action is performed. After testing, the effect is better when the area threshold is set to 24000 and the central area deviation range is set to 50.

E. Capturing objects

After obtaining the position information of the target in the image, it is necessary to control the steering gear to make the manipulator move, drive the manipulator to approach and grab the target, and reset after successful grabbing. The Raspberry Pi controller sends instructions to the 16-channel steering gear control board through serial communication, so as to control the rotation speed and rotation angle of the steering gear. The rotation speed can be fixed, so only the angle control variables need to be converted into hexadecimal numbers according to the communication protocol.The citrus is placed in the manipulator. At this time, it is the grasping state. The center coordinates (xc, yc) and size information (w, h) of the circumscribed rectangle of the citrus are obtained when the manipulator is grasping. The (xc, yc, w, h) is used as the threshold so that the mechanical arm can drive the manipulator to approach the target, and finally, the threshold range is reached to perform the grasping action.

It is assumed that steering gear 1 controls joint 1, steering gears 2 and 3 control joint 2, and steering gears 4,5 and 6 control joints 3,4 and 5 in Figure 2. To make the manipulator approach the manipulator continuously, it is necessary to judge whether the grasping state is reached according to the relationship between the position and size of the citrus and the threshold.The judgment logic is as follows: if the abscissa of the center of the circumscribed rectangle of the citrus is inclined left or right, then joint 1 turns left or right by 1°; if the rectangular area is less than the graspable threshold, joint 2 is controlled to make the manipulator decrease by 1°, otherwise, it increases by 1°. If the citrus center is inclined upper, joint 3 is controlledto make the manipulator rise by 1°. To keep the manipulator horizontal, joint 4 controls the manipulator to reduce by 1°.If the citrus center is lower, the opposite is true.When the circumscribed rectangle of the citrus image reaches the threshold range, the movement of other manipulator joints is immediately stopped, and only joint 5 is controlled to make the manipulator perform the grasping action. An infrared ranging sensor is usedto judge whether it can be grasped more accurately. When the target is detected by the infrared ranging sensor, a reference grasping signal is also given to assist the grasping judgment.

After carrying out the grasping action, the mechanical arm and the manipulator are controlled to a fixed position. The manipulator is opened, and the grasped citrus is placed behind the vehicle body. Then the whole manipulator is reset and returns to a non-grasping state, that is, a grasping cycle is completed.

IV. Conclusion

The smart picking vehicleis characterized by low cost, easy operation and good picking effect, and it generally meets the design requirements. Besides, it is an attempt to realize agricultural automatic picking, and there is still much room for improvement.

References

[1] Chen Xiaoyu, Ma Xinglu, Liu Yang. Research on the Action of a Robot Arm for Picking Sainte Maiden Fruit Based on Machine Vision [J]. Computer Measurement and Control, 2024 (002): 032

[2] Zhou Yunshan, Li Qiang, Li Hongying, et al. Application of Computer Vision on Mushroom Picking Robots [J]. Journal of Agricultural Engineering, 1995, 11 (4): 6. DOI: 10.1007/BF02943514

[3] Cui Peng, Chen Zhi, Zhang Xiaochao. Static analysis and simulation of bionic robotic arms for apple picking [J]. Journal of Agricultural Machinery, 2011. DOI: CNKI: SUN: NYJX. 02011-02-032

[4] Han Kun, Peng Jingying. Research on pedestrian multi-target tracking algorithm based on improved YOLOX and multi-level data association [J]. Journal of Railway Science and Engineering, 2024, 21 (1)

[5] Meng Jianjun, Chen Xiaotong, Li Decang, et al. Pose estimation processing methods in computer vision technology [J]. Computer Simulation, 2023, 40 (5): 274-278