IJISET - International Journal of Innovative Science, Engineering & Technology, Vol. 4 Issue 2, February 2017
ISSN (Online) 2348 – 7968 | Impact Factor (2016) – 5.264
www.ijiset.com

Implementation of Leap Motion Based RC Car Controller

Ye Hoon Lee

Department of Electronic & IT media Engineering, Seoul National University of Science & Technology, Seoul, 01811, Korea

Abstract

In recent years, there has been a limitation of the touch screen, for example, when the user tries to use the touch screen and falls down on the road, and thus a new input interface is required. An alternative is the *Leap Motion*, which has been developed to replace the existing interface and to support the 3D environment in particular by providing ultra-high resolution and fast recognition performance at an affordable price. In this paper, we consider the Leap Motion controlling the RC car through this technique. The motivation is that it is much easier to navigate through the existing joystick or smartphone application. Although the Leap Motion technology has not been commercialized so far, it is obvious that it is the technology that will lead the future. Therefore, we implement a RC car controller using Leap Motion technology in order to contact with and study the technology first.

Keywords: *Leap Motion, RC Car, Controller, Technology, Interface.*

1. Introduction

Input interfaces have evolved through keyboards and mouse for desktop computing, and touchscreens for mobile computing. However, in recent years, there has been a limitation of the touch screen such as a case where the user tries to use the touch screen and falls over the road, and a new input interface is required. An alternative is the *Leap Motion*. Lip Motion is an input device that can recognize fingers and other objects with a precision of 1/100mm. A small iPod-sized device reads very precisely the cubic feet of three-dimensional space. Leap Motion has been developed to replace the existing interface and to support the 3D environment in particular by providing ultra-high resolution and fast recognition performance at an affordable price.

Lip Motions are attractive because they are more colorful and relatively new to the other controllers, joysticks and mobile phone applications. Indeed, in December 2012, MIT's Technology Reviews, an engineering review by MIT, introduced lip motion as "the most important new technology ever since a smartphone." Although the Leap Motion technology has not been commercialized so far, it is obvious that it is the technology that will lead the future. Therefore, we decided to implement a RC car controller using Leap Motion technology in order to contact with and study the technology first.

There are two main goals in this paper. The first goal is to implement an operational RC car through a microprocessor. This allows us to better understand the basic functions of the processor and control the RC cars. The boards used are Raspberry Pi 2 and STM32f051 boards. The second goal is to replace the existing controller with a Lip Motion device. That is, the basic operations of the car (forward, backward, left turn, right turn) are controlled by using a hand operation rather than a joystick or an application. To this end, we will examine the programming of Lip Motion in detail in the followings.

2. System Model and Design

The block diagram of the system is as shown in Fig. 1. The input part, Leap Motion, and the laptop are wired. Raspberry Pi 2 and STM32 board communicate with each other wirelessly (using socket programming), and Raspberry Pi 2 and STM32 board communicate with each other through serial peripheral interface (SPI)). Raspberry Pi 2 and STM32 use camera module and motor control. It is connected to the motor drive.

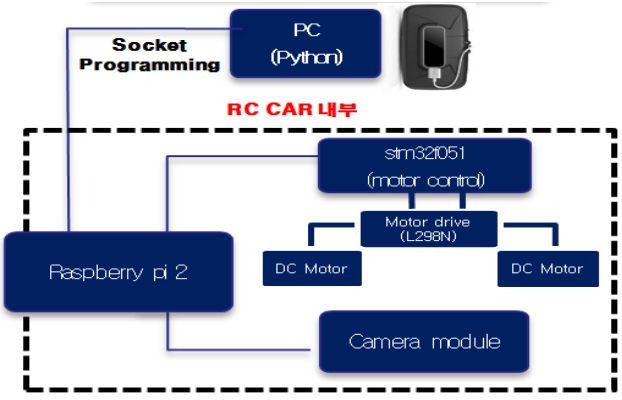

Fig. 1 System block diagram.

It is used in conjunction with laptops due to the nature of Lip Motion devices, and RC cars are manufactured using RC car kits. In addition, the camera module of Raspberry Pi 2 is mounted on the RC car. It is designed to control the motor of the RC car according to the input because the lip motion device which is different from the conventional joystick is used. We also design a way to use the camera features with the camera module of Raspberry Pi 2. As a result, three motions (data) of Lip Motion are used to control the motor and implement the camera function. Depending on the nature of the Lip Motion, the various hand gestures used in the work are designed to be applied instantly through software modifications. RC cars are designed for additional functions such as camera functions as well as driving purposes.

The S/W operation of the input part uses the application programming interface (API) of Leap Motion. There are three types of input data. We implement motor control and camera functions using hand height, tilt, and circle motion. In Raspberry Pi 2, circle motion data is controlled to adjust the camera module, and SPI communication is enabled by modifying the values of the remaining two data (using image transmission/reception socket programming). The STM32f051 is employed to control the motor by converting incoming data into pulse width modulation (PWM) signal.

3. System Implementation

3.1 Hardware Module

Fig. 2 The sensor part (Raspberry pi 2).

H/W can be roughly divided into an input section, a sensor section, and an operation section. The input part, Leap Motion device and laptop, are connected only if there is Leap Motion program. The sensor part is the Raspberry Pi 2 area, which recognizes the data of the input Leap Motion part and passes the value matching the condition to the operation part. The function of operating the camera module in the Rapsberry Pi 2 is performed separately from the operation unit. The operation method of the motor control part of the RC car is selected as the *steering method* by the wheel speed.

Fig. 3 Operation part (STM32f051 - Motor).

3.2 Software Algorithm

If you look at the schematic S/W algorithm (also divided into input, sensor, and operation), input of Leap Motion API (hand, X, Y, Z slope, motion of circle) is passed to Raspberry Pi 2 using TCP/IP communications. In the sensor part, motion data is changed to positive value for smooth SPI communication. Thereafter, the SPI communication is performed with operation unit of STM32f051. Also, when the circle data is input, the camera module is activated to capture the image and generate an image file. This image file is transferred to the laptop connected to the Leap Motion part again, and shown on the screen.

To control the motor through the Leap Motion hand, the slope values of X, Y, Z-axis tilt data is used to activate the forward and backward movement. The slope data with the X-axis is used to control the left turn and the right turn. Also, in case of data error, we set the first 10 data to move correctly after entering the correct data to prevent the RC car from malfunctioning.

4. Performance Test and Evaluation

In order to operate the whole RC car, all parts in the input part, the sensor part, and the operation part should operate normally. The parts that have largely confirmed operation are Leap Motion of the input part, Raspberry Pi 2 and STM32f051 of the sensor part, L298N (motor drive) of the operation part, and the motor and the camera module connected thereto.

Check Leap Motion Device: You can check whether APP recognizes your hand normally through Leap Motion homepage.

Fig. 4 Check Leap Motion device error.

Socket Programming Confirmation: It is confirmed that data of one row and three columns received through Leap Motion through server (right) and client (left) are smoothly transmitted and received.

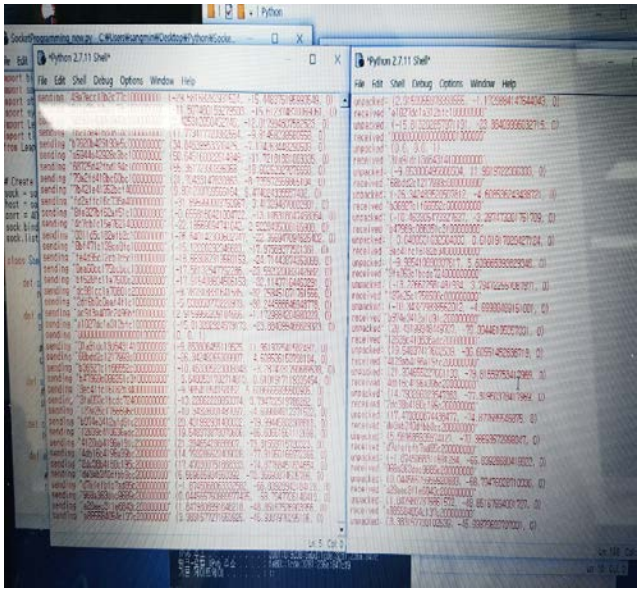

Fig. 5 Socket programming data transmission and reception confirmation.

Check Camera Module Operation: Recognize circle gesture in Leap Motion, check camera module operation, picture transfer and file transfer.

Fig. 6 Camera module operation.

Motor Control through Leap Motion: By using data of Leap Motion, forward and reverse operation of RC cars is possible. When the hand is tilted forward, the RC car moves in the forward direction and when the hand is tilted back it moves in the reverse direction, which utilizes the Y-axis tilt of Leap Motion. If you tilt your hand to the right by tilting the X-axis, it increases the speed of the left wheel to make a right turn. If you tilt to the left, it increases the speed of the right wheel to make a left turn. When the circle gesture is recognized on the Leap Motion, the camera module connected to the Raspberry Pi 2 takes the picture and sends it to the PC.

Fig. 7 Motor control through Leap motion.

4. Conclusions

In this paper, we made an implementation of RC cars and examined that software and hardware are properly combined to create a finished RC car prototype. It was observed that if the socket programming and the SPI communication were smooth, the motor would run, but the data was broken by the back electromotive force or the high frequency signal generated by the motor. Therefore, it was possible to design a filter stage using a capacitor to achieve smooth communication. In this way, we realized that if the object to be controlled through the Leap Motion is an electric motor, it generates noise, and therefore, a filter must be manufactured through a sufficient considerations.

The Leap Motion is expected to be a novel input device rather than a manipulation with existing joystick or smartphone application. In addition, Leap Motion and VR can work together to create a more realistic RC car. The Leap Motion is not limited to devices, so it can be applied to large-sized devices such as airplanes and automobiles in the long run. Current RC cars can move within the range of the mobile phone or Wi-Fi, but if we solve the problem with this possibility or attach the EGG to the RC car network, the application of Leap Motion can be extended to the airplane or car in the future.

Acknowledgments

This study was supported by the Research Program funded by the Seoul National University of Science and Technology (2016-1695).

References

[1] S. Yoon, TCP/IP Socket Programming, Korea: Orange Media, 2009.
[2] S. A. Rago, R. W. Stevens, Advanced Programming in the UNIX Environment, -Wesley Professional, 2008.
[3] J. Catsoulis, Designing Embedded Hardware, Oreilly, 2002.
[4] D. Y. Yoon, ARM Cortex-M0 STM32F051 Master, OHM, 2013.

Ye Hoon Lee received the Ph.D. degrees in electrical engineering from Korea Advanced Institute of Science and Technology (KAIST), Daejeon, Korea, in 2000. Since 2005, he has been with the Department of Electronic & IT Media Engineering, Seoul National University of Science and Technology, Seoul, Korea. His research interests include wireless communications emphasis on adaptive resource allocations and the baseband transceiver design in wireless channels.