

Bluetooth control Car Obstacle Avoidance Car

Tsang Chun Leung 54802143
Yiu Pui Kwan 54811987

What has been done

Car with two operating modes :

We have written an app for the Bluetooth connection to the car.

Two modes can be selected and changed within the app.

a. Automatic detection-

detects its distance from the surrounding objects.

If the distance < 50cm, turn to other direction to avoid it.

b. Remote control via Bluetooth connection-

Players can operate the car by controlling its moving direction with the app after connecting the smartphone to the car by Bluetooth.

Tools:

L298n motor controller

HC-05 Bluetooth Serial Module

Arduino UNO

Motor x 2

Ultrasonic Sensor

3.7V battery

Other:

Wheel x 2

wooden plate x 2

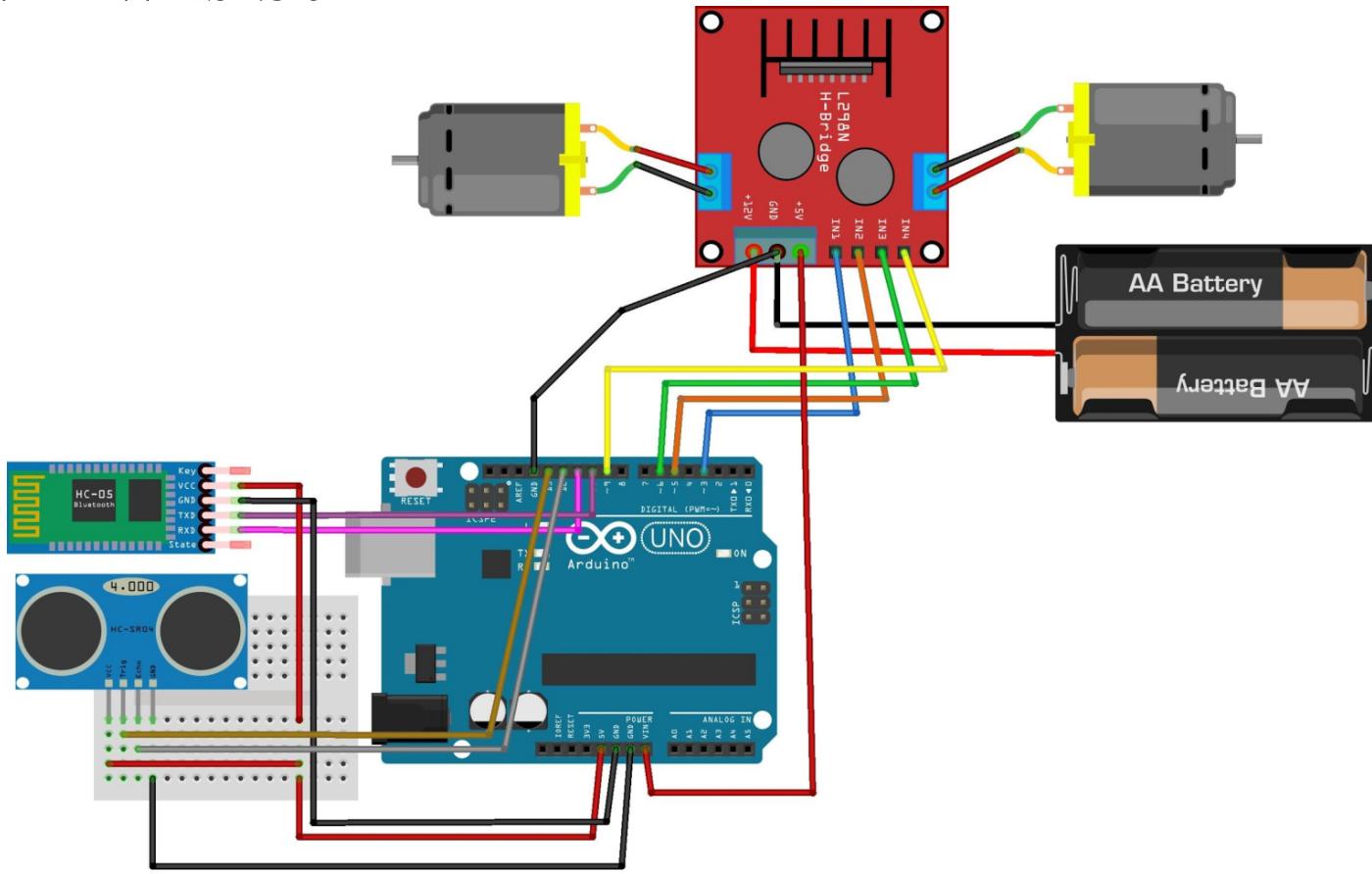
Software:

Arduino

Appinventor

Android system

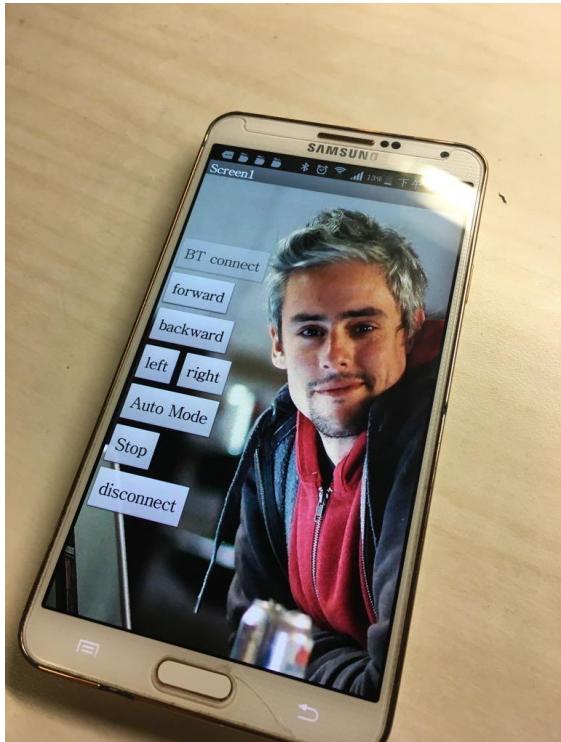
How it was built



App(use MIT App Inventor)

We created an App for controlling the car on

<http://ai2.appinventor.mit.edu/?locale=en#6069953337491456>



Viewer

Display hidden components in Viewer
 Check to see Preview on Tablet size.

Screen1

BT connect

forward

backward

left right

Auto Mode

Stop

disconnect

9:48

Non-visible components

BluetoothClient1

Components

- Screen1
 - ListPicker1
- VerticalArrangement1
 - Button1
 - Button2
- HorizontalArrangement1
 - Button3
 - Button4
 - Button5
 - Button6
- disconnect
- BluetoothClient1

Rename Delete

Media

krahd_13...26_03.jpg

Upload File ...

when Screen1 .Initialize

do

- set ListPicker1 . Enabled to true
- set Button1 . Enabled to false
- set Button2 . Enabled to false
- set Button3 . Enabled to false
- set Button4 . Enabled to false
- set Button5 . Enabled to false
- set Button6 . Enabled to false
- set disconnect . Enabled to false

when ListPicker1 .BeforePicking

do

- set ListPicker1 . Elements to BluetoothClient1 . AddressesAndNames

when ListPicker1 .AfterPicking

do

- if call BluetoothClient1 . Connect

address ListPicker1 . Selection

then

- set ListPicker1 . Enabled to false
- set Button1 . Enabled to true
- set Button2 . Enabled to true
- set Button3 . Enabled to true
- set Button4 . Enabled to true
- set Button5 . Enabled to true
- set Button6 . Enabled to true
- set disconnect . Enabled to true

when Button1 ▾ .TouchDown
do call BluetoothClient1 ▾ .SendText
text “ a ”

when Button2 ▾ .TouchDown
do call BluetoothClient1 ▾ .SendText
text “ b ”

when Button3 ▾ .TouchDown
do call BluetoothClient1 ▾ .SendText
text “ c ”

when Button4 ▾ .TouchDown
do call BluetoothClient1 ▾ .SendText
text “ d ”

when Button5 ▾ .Click
do call BluetoothClient1 ▾ .SendText
text “ t ”

when Button1 ▾ .TouchUp
do call BluetoothClient1 ▾ .SendText
text “ s ”

when Button2 ▾ .TouchUp
do call BluetoothClient1 ▾ .SendText
text “ s ”

when Button3 ▾ .TouchUp
do call BluetoothClient1 ▾ .SendText
text “ s ”

when Button4 ▾ .TouchUp
do call BluetoothClient1 ▾ .SendText
text “ s ”

when Button6 ▾ .Click
do call BluetoothClient1 ▾ .SendText
text “ s ”

```
when disconnect .Click
do call BluetoothClient1 .Disconnect
  set ListPicker1 . Enabled to true
  set Button1 . Enabled to false
  set Button2 . Enabled to false
  set Button3 . Enabled to false
  set Button4 . Enabled to false
  set Button5 . Enabled to false
  set Button6 . Enabled to false
  set disconnect . Enabled to false
```

Arduino Code:

```
int trig = 13;
int echo = 12;
int in1 = 3;
int in2 = 5;
int in3 = 6;
int in4 = 9;
SoftwareSerial I2CBT(10,11);

void setup() {
    Serial.begin(9600);
    I2CBT.begin(9600); //bluetooth baud rate
    pinMode(trig, OUTPUT);
    pinMode(echo, INPUT);
    pinMode(in1, OUTPUT);
    pinMode(in2, OUTPUT);
    pinMode(in3, OUTPUT);
    pinMode(in4, OUTPUT);
}
```

case 97:

```
digitalWrite(in1,HIGH);
digitalWrite(in2,LOW);
digitalWrite(in3,LOW);
digitalWrite(in4,HIGH);
break;
```

case 98:

```
digitalWrite(in1,LOW);
digitalWrite(in2,HIGH);
digitalWrite(in3,HIGH);
digitalWrite(in4,LOW);
break;
```

```
case 116:  
digitalWrite(trig, LOW);  
delayMicroseconds(2);  
digitalWrite(trig, HIGH);  
delayMicroseconds(5);  
digitalWrite(trig, LOW);  
int duration = pulseIn(echo, HIGH);  
int cm = duration / 29 / 2;  
if(cm>50){  
    digitalWrite(in1,HIGH);  
    digitalWrite(in2,LOW);  
    digitalWrite(in3,LOW);  
    digitalWrite(in4,HIGH);  
        1000 ms
```

Why it is interesting

- **Playful**
conducting competition or game
- **Interactive**
interacts with the environment : sensing and avoiding objects around it
interacts with humans : controlling its direction to move.
- **Two modes for selection**
Even only one player can compete with other cars with obstacle avoiding mode.

Improvement and Future works

- Automatic detection part**

To detect obstacle from different sides but not only from the front side:

- set up the Ping which can rotate by 360 degrees with a stepper motor
 - detect surrounding objects around the car in order to avoid hitting them

- Sensing light for acceleration**

Set a LDR light sensor under the car

- When the car is passing the lighting part of the road and LDR senses it, the motors are triggered to rotate faster for acceleration.

Reference(Process Reference)

<http://www.appinventor.tw/whatis/>

[雙A計劃] Part0 : App Inventor 透過藍牙傳送訊號給 Arduino

<http://blog.cavedu.com/programming-language/appinventor/appinventorandarduinowithbluetooth/>

[雙A計劃] Part1 : App Inventor 經由藍牙控制 Arduino LED 亮滅

<http://blog.cavedu.com/programming-language/appinventor/%E9%9B%99a%E8%A8%88%E5%8A%83-part1%EF%BC%9Aapp-inventor-%E7%B6%93%E7%94%B1%E8%97%8D%E7%89%99%E6%8E%A7%E5%88%B6-arduino-led-%E4%BA%AE%E6%BB%85/>

Reference:

Learn how to use L298N

<https://www.youtube.com/watch?v=kv-9mxVaVzE>

Learn to use app inventor(russian website)

<https://geektimes.ru/post/255442/>

