

通訊網路實驗 Lab4 Report

110511254 徐煜絨

一、實驗目的（第二部分實驗說明會有更詳細介紹）

Q1

修改 turtlebot3_teleop_key.py，加上使其能左右移動的程式碼。

Q2

以 Android Studio APP 控制 turtlebot 前後移動。

Q3

結合以上兩題，並在 APP 中加上往左和往右的按鈕，並在 turtlebot_app.py 加上對應的移動程式，使 turtlebot 可以全方位移動。

二、實驗過程（Code+說明）

Q1

```
# 前進
if key == 'w' :
    # 線性速度是原先的速度加上單位速度，不改變 angular velocity
    target_linear_vel= checkLinearLimitVelocity(target_linear_vel+ LIN_VEL_STEP_SIZE)
    status = status + 1
    print(vels(target_linear_vel,target_angular_vel))
# 後退
elif key == 'x' :
    target_linear_vel= checkLinearLimitVelocity(target_linear_vel- LIN_VEL_STEP_SIZE)
    status = status + 1
    print(vels(target_linear_vel,target_angular_vel))
# 左邊
elif key == 'a' :
    # 角速度是原先的角速度加上單位角速度，不改變 linear velocity
    target_angular_vel=checkLinearLimitVelocity(target_angular_vel+ANG_VEL_STEP_SIZE)
    status = status + 1
    print(vels(target_linear_vel,target_angular_vel))
# 右邊
elif key == 'd' :
    target_angular_vel=checkLinearLimitVelocity(target_angular_vel-ANG_VEL_STEP_SIZE)
    status = status + 1
    print(vels(target_linear_vel,target_angular_vel))
```

```
# 停止，linear 和 angular 速度都歸零
elif key == ' ' or key == 's' :
    target_linear_vel = 0.0
    control_linear_vel = 0.0
    target angular_vel = 0.0
    control angular_vel = 0.0
    print(vels(target_linear_vel, target angular_vel))
```

A 和 D 表示左右轉，會讓原先的 target angular_vel 減掉 ANG_VEL_STEP_SIZE 得到新的旋轉程度（越轉越快或越慢）。

Q2

```
#socket
HOST = '192.168.0.135'
PORT = 8001
try:
    sock = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
except socket.error, msg:
    sys.stderr.write("[ERROR] %s\n" % msg[1])
    sys.exit(1)

try:
    sock.bind((HOST,PORT))
    sock.listen(5)
except socket.error, msg:
    sys.stderr.write("[ERROR] %s\n" % msg[1])
    exit(1)
```

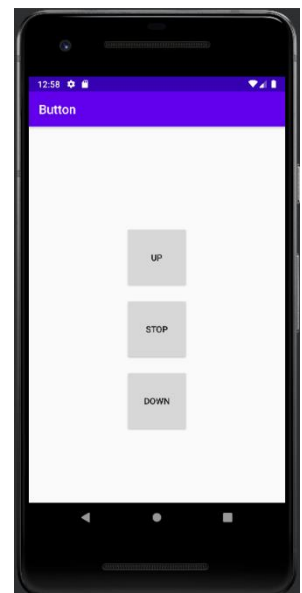
TCP connect

綁定 socket 的 IP

Port 的最大連線數

由上可以知道 HOST 是 192.168.0.135，使用 TCP connect，並且 port 是 8001。

```
protected Void doInBackground(String... params){
    try{
        try{
            Socket socket = new Socket( host: "192.168.0.135", port: 8001);
            PrintWriter outToServer= new PrintWriter(
                new OutputStreamWriter(
                    socket.getOutputStream()));
            outToServer.print(params[0]);
            outToServer.flush();
        }catch (IOException e){
            e.printStackTrace();
        }
    }catch (Exception e)
    {
        this.exception = e;
        return null;
    }
    return null;
}
```



把 host 和 port 設為和 turtlebot_app 中相同。

Q3

```
button_left.setOnClickListener(new Button.OnClickListener(){

    @Override

    public void onClick(View v) {
        System.out.println("left");
        new SendData().execute("a");
    }

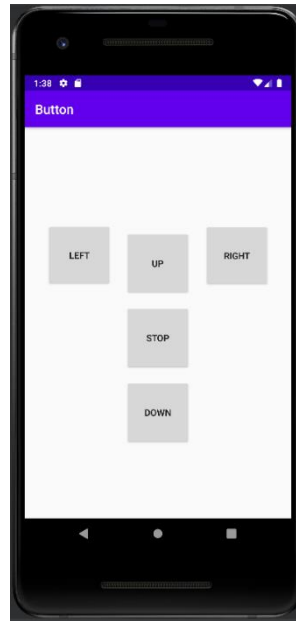
});

button_right.setOnClickListener(new Button.OnClickListener(){

    @Override

    public void onClick(View v) {
        System.out.println("right");
        new SendData().execute("d");
    }

});
```



在原先的 layout 上新增往左和往右的 button，在 xml 檔中完成兩個按鍵的指令。

三、問題及解法

我一開始在第三題加按鈕並執行時，手機畫面上沒有出現 LEFT 和 RIGHT 兩個按鈕，後來詢問學長姐發現有人也遇過這樣的錯誤，我原本拉了三個方向固定按鈕（正 x、正負 y），改成在 x 和 y 各只拉一個固定就成功在手機上顯示按鈕。雖然顯示了但位置卻不如預期，這部分目前還沒找到解法。

四、心得

這次實驗在 ROS 上完成機器控制，我們練習了對個別 node 要怎麼給予指令，並讓他能成功完成動作，也練習 master 之間的溝通，綜合並實現對跨網的知識。雖然在 layout 上有顯示問題，但這次實驗還是讓我有所收穫！