Build a segway vehicle that can roam around obstacles on the floor of the lab

Tested three different approaches to build a segway that can roam around obstacles on the floor. The three approaches are as follows:

	Segway with one light sensor
	Segway with two light sensors
	Segway with Gyro sensor

Table 1 : Tested Approaches

1) Segway with one light sensor

The objective is to use the light sensor is to track the approximate distance to the floor. Figure 1 depicts the segway with one light sensor.

The problem is keeping the segway in a range of light values reding by light sensor in such a way that it could the segway to get balanced. The observed issue with this method is the threshold which could happen in two different distances. The segway uses PID algorithm to balance. Table 2 shows the balancing function in C language.

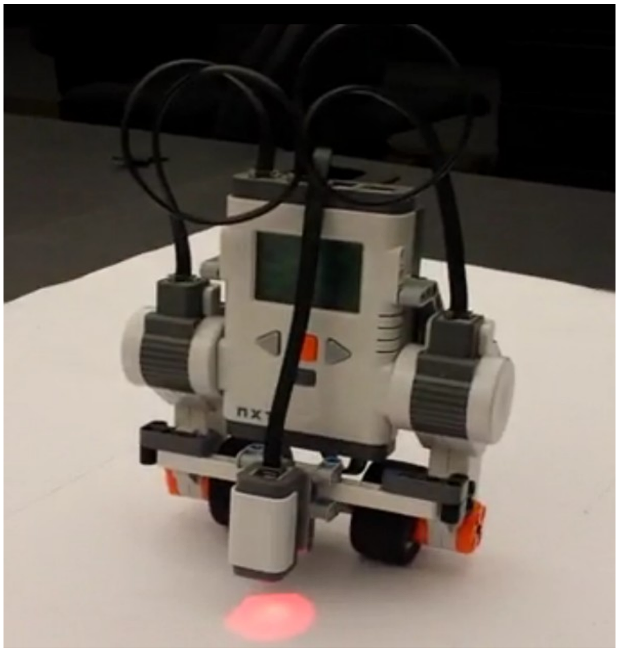


Figure 1: Segway with one Light Sensor

```
/* Proportional Error:*/
int error = current_color  - LIGHT_THRESHOLD;

/* Integral Error:*/
integral_error = ((integral_error + error) * 2)/3;

/* Derivative Error: */
int derivative_error = error - prev_error;

prev_error = error;

int pid = (int)(Kp * error + Ki * integral_error + Kd * derivative_error) / scale;

if (pid > 100)
    pid = 100;
if (pid < -100)
    pid = -100;

/* Power derived from PID value:
 abs (pid)*/
int power = pid * ( (pid<0) * (-1) + (pid>0));
power = 55 + (power * 45) / 100; // NORMALIZE POWER

if (pid > 0){ /*FORWARD*/
     nxt_motor_set_speed(RIGHT_MOTOR, power, 1);
     nxt_motor_set_speed(LEFT_MOTOR,  power, 1);
} else { /*BACKWARD*/
     nxt_motor_set_speed(RIGHT_MOTOR, -power, 1);
     nxt_motor_set_speed(LEFT_MOTOR,  -power, 1);
}
```

Table 2: Balancing Function with one Light Sensor in C

2) Segway with two light sensors
To fix the problem of double ranges, the segaway uses two light sensors. However, the result shows because of the sensitivity of light sensors, balancing the segway is still not feasible.

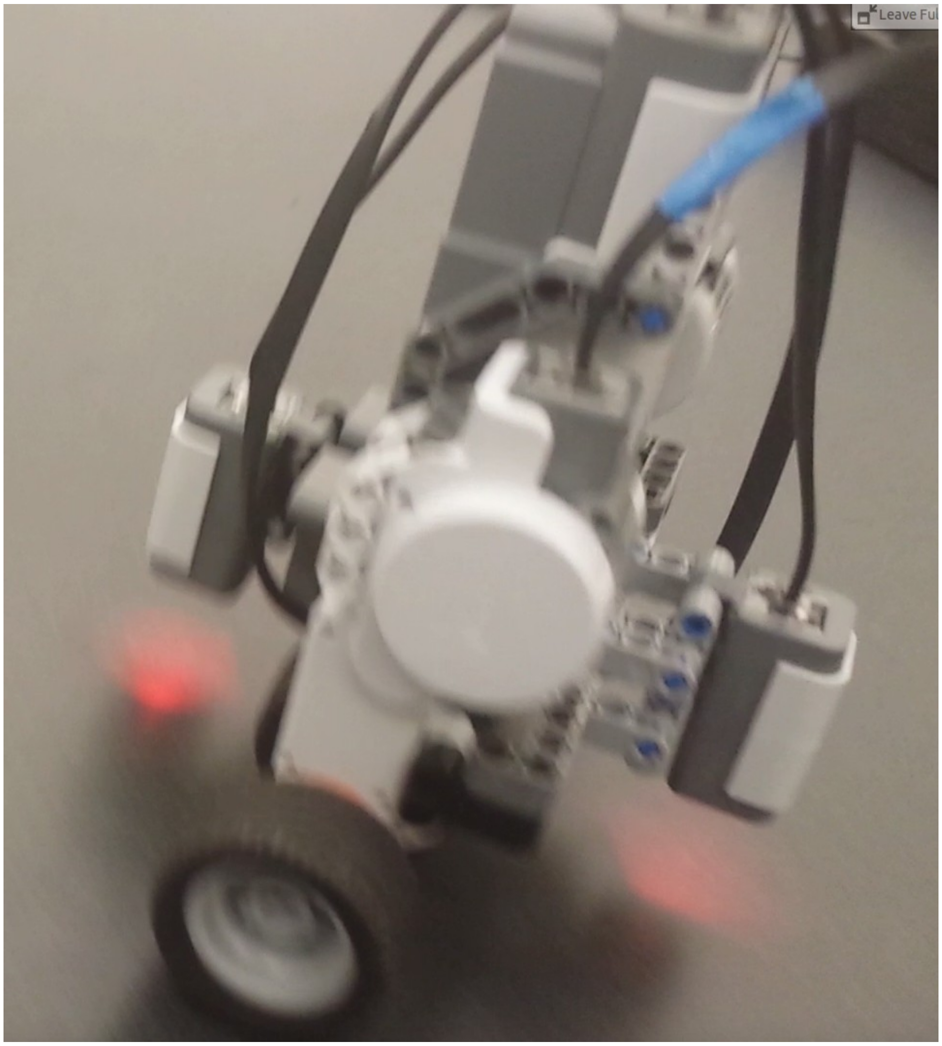


Figure 2: Segway with two Light Sensors

```
/* Proportional Error:*/
    int error = (current_color_front - (current_color_rear))/2;
```

Table 3: Proportional Error Calculation with Two Light Sensors

3) Segway with Gyro sensor

This segway utilizes the NXT Gyro sensor. This sensor contains a single axis gyroscopic sensor that detects rotation and returns a value representing the number of degrees per second of rotation [1]. A library called *balance_control* is provided in “***lejos-osek/nxtOSEK/ecrobot/nxtway_gs_balancer”*** directory.

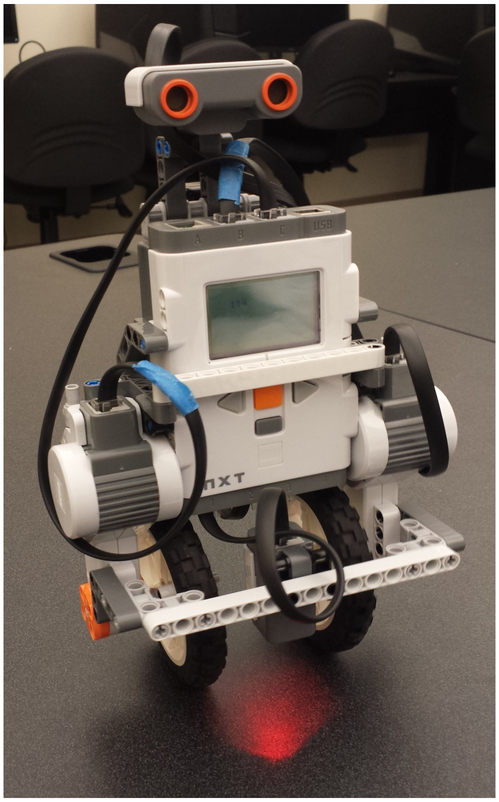


Figure 3: Segway with one Gyro Sensor

```
balance_control(
        (F32)cmd_forward,
        (F32)cmd_turn,
        (F32)ecrobot_get_gyro_sensor(PORT_GYRO),
        (F32)gyro_offset,
        (F32)nxt_motor_get_count(PORT_MOTOR_L),
        (F32)nxt_motor_get_count(PORT_MOTOR_R),
        (F32)ecrobot_get_battery_voltage(),
        &pwm_l,
        &pwm_r);
      nxt_motor_set_speed(PORT_MOTOR_L, pwm_l, 1);
      nxt_motor_set_speed(PORT_MOTOR_R, pwm_r, 1);
```

Table 4: NXTway-GS C API balance control function [2]

This segway has this capability to detect different surfaces by using a light sensor. Fig. 4 shows that when it detects white surface, it does turn left.

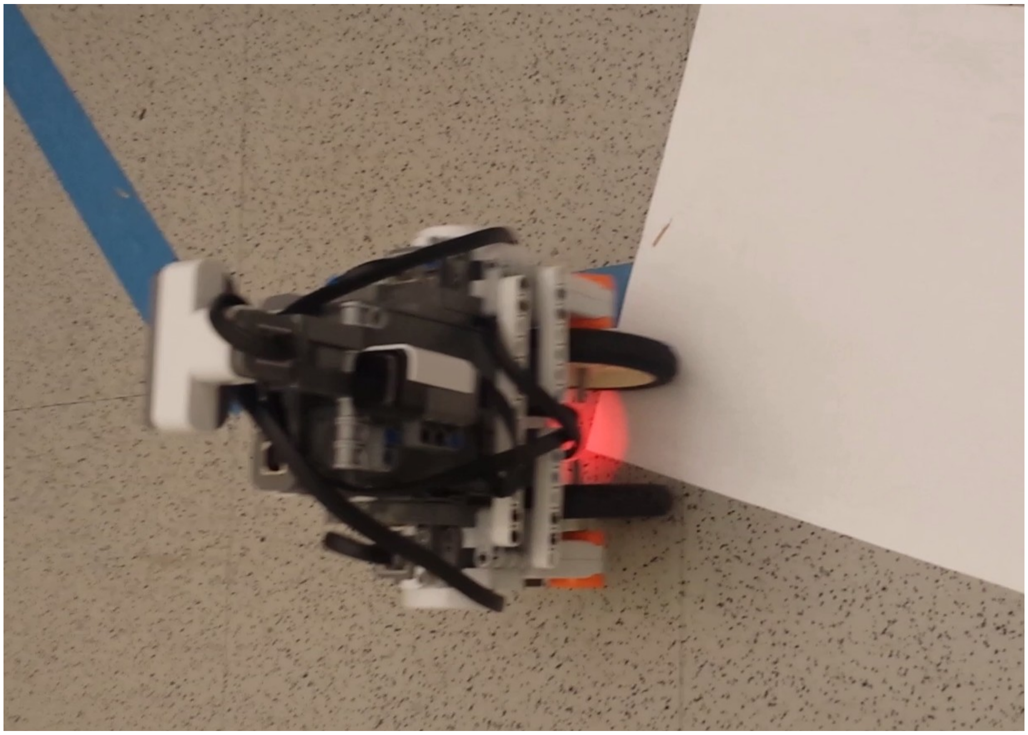
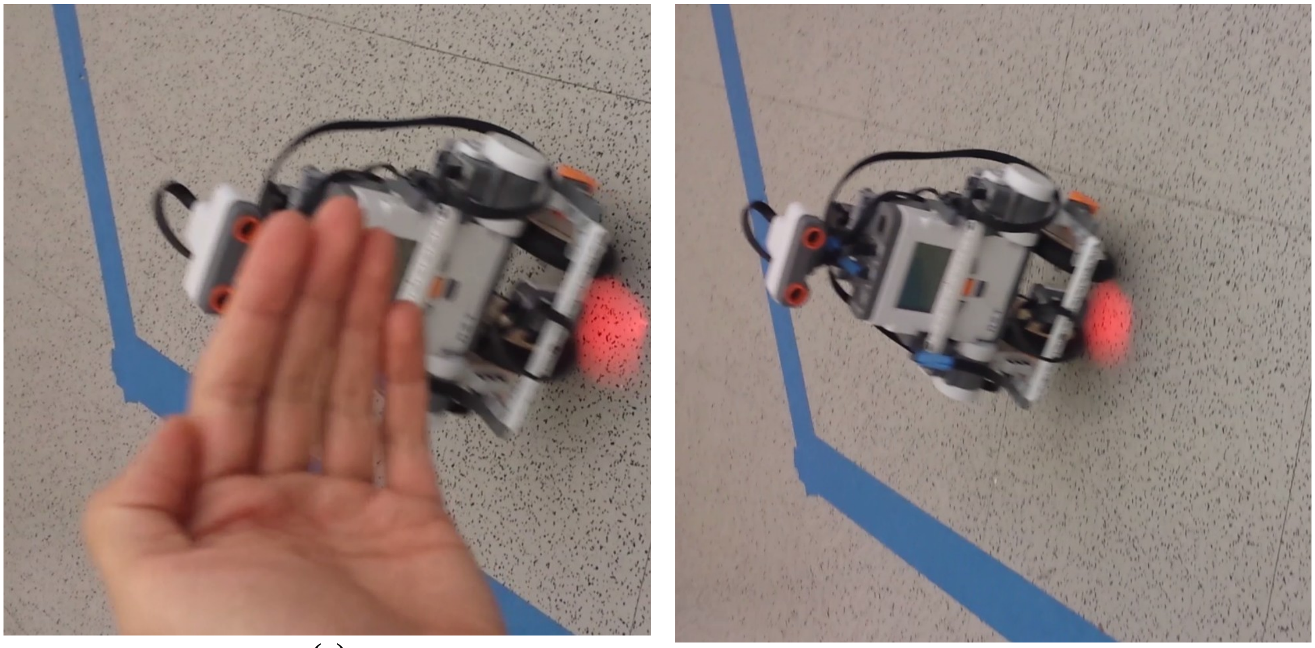


Figure 4: Finding different surfaces using one Light Sensor

Fig. 5 shows object detection using a sonar sensor. The segway goes backward when it detects an object, and quickly balances itself.



(a)

(b)

Figure 5: Object detection using sonar sensor

[1] https://www.hitechnic.com/cgi-bin/commerce.cgi?preadd=action&key=NGY1044
[2] http://lejos-osek.sourceforge.net/nxtway_gs.htm