Leap Motion + Cylon.js 控制 Sphero

Phodal Huang

October 24, 2017

目录

在我业余的时间里，我喜欢玩硬件和机器人。我是从 Node.js 开始的，但是最近我发现了 Cylon.js，并且在快速试玩之后，发现它非常棒，并决定用这个框架重写我的项目。

作为起点，我决定重新编写项目，用 Leap Motion Controller 控制 Sphero。

您可以在这里找到原始的代码：Leap Motion Sphero，但以下是几个代码段：

```
1 module.exports = function() {

2

3   var Leap = require('leapjs');

4   var spheron = require('spheron');

5

6   // Set this to the device Sphero connects as on your computer.

7   var device = '/dev/tty.Sphero-RBR-AMP-SPP';

8

9   var safeMode = true; //Turn this off if Sphero is in water or you like to
        live dangerously!

10

11  var controlSphero = function(sphero) {

12    var controller = new Leap.Controller({frameEventName:'deviceFrame',
          enableGestures:true});

13    controller.on('connect', function() {

14      console.log('connected to leap motion');

15    });

16    controller.on('ready', function() {

17        console.log('ready');

18    });
```
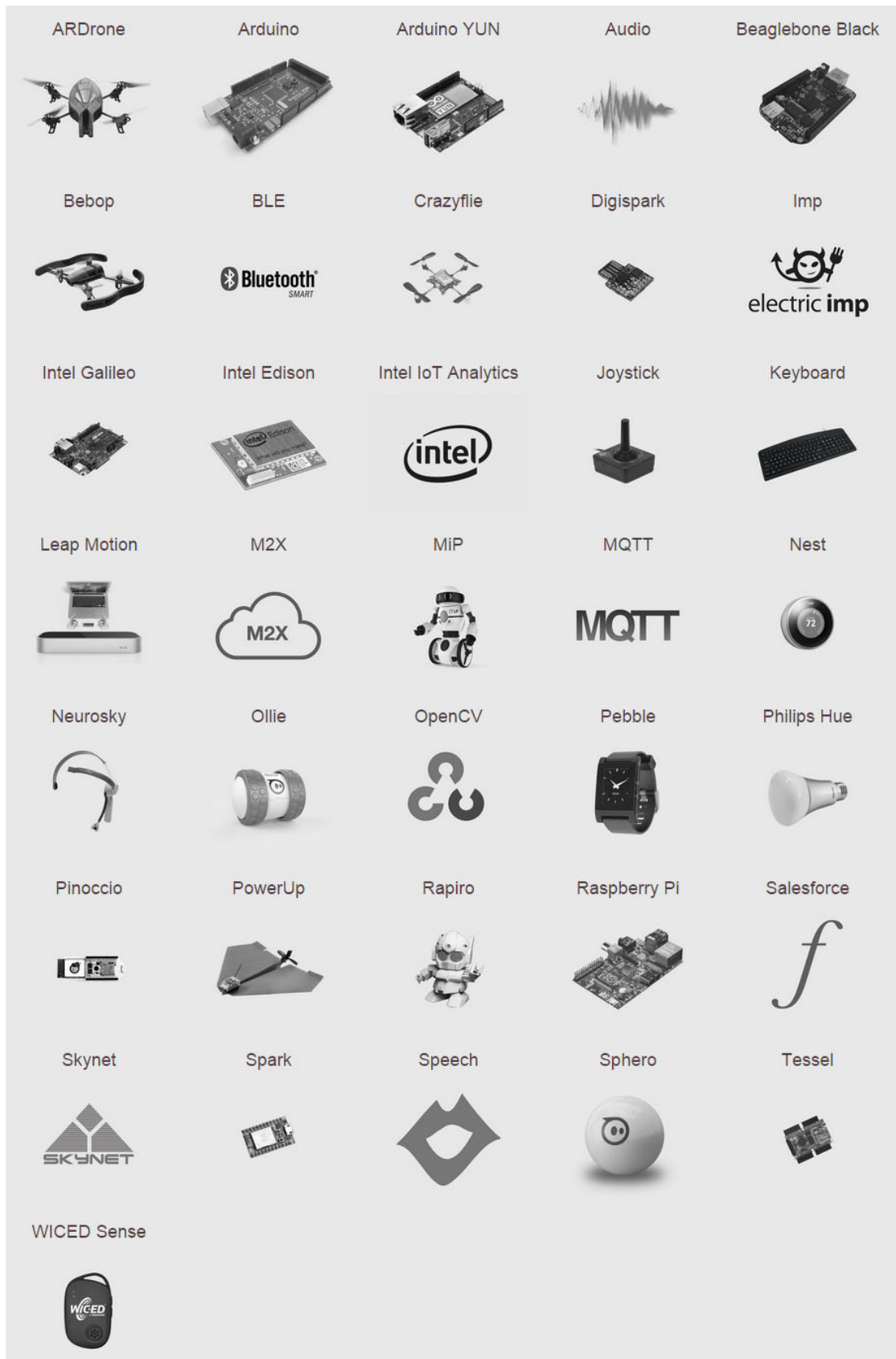
```
1 switch (direction) {

2   // Original settings included: 'sphero.heading = (heading value)';

3   // Original speed for all of them: 128.

4   case 'LEFT':

5     //sphero.roll(speed, heading, state, option)

6     sphero.roll(70, 270, 1); //Heading is expressed in degrees so 270 will
          make the ball move to the left.

7     break;

8   case 'RIGHT':
```

```
 9      sphero.heading = 90;
10 sphero.roll(70, 90, 1);      break;
11   case 'UP':
12     stopSphero(sphero);
13     //Make the ball turn blue when users move their hand up.
14     ball.setRGB(spheron.toolbelt.COLORS.BLUE).setBackLED(255);
15     break;
16   case 'DOWN':
17     stopSphero(sphero);
18     //Make the ball turn white when users move their hand down.
19      ball.setRGB(spheron.toolbelt.COLORS.WHITE).setBackLED(255);
20     break;
21   case 'FORWARD':
22      sphero.roll(70, 0, 1);
23     break;
24   case 'REVERSE':
25     sphero.heading = 180;
26     sphero.roll(70, 180, 1);
27     break;
```

它的工作方式非常简单。Sphero 通过蓝牙连接，Leap Motion 控制器需要插入计算机。当探测到 Sphero 后，Leap Motion 设备将追踪手的动作，方向将在 Sphero 上。你可以查看 Github 上的代码。

现在，我们来看看 Cylon.js。我最先注意到这个框架是因为，为了获得相同结果所需的代码量很少。完成同样的事情，我只需要 68 行代码！

我想让它使得更容易的原因是，Cylon 已经有一些现有的模块，可以在某些设备进行编程，如下所示：

ARDrone	Arduino	Arduino YUN	Audio	Beaglebone Black
Bebop	BLE	Crazyflie	Digispark	Imp
Intel Galileo	Intel Edison	Intel IoT Analytics	Joystick	Keyboard
Leap Motion	M2X	MiP	MQTT	Nest
Neurosky	Ollie	OpenCV	Pebble	Philips Hue
Pinoccio	PowerUp	Rapiro	Raspberry Pi	Salesforce
Skynet	Spark	Speech	Sphero	Tessel
WICED Sense				

要开始使用 Cylon，您需要引入（require）它，并指定您正在使用的设备。

```
1 var Cylon = require('cylon');
2 Cylon.robot({  connections: {
3     leapmotion: {adaptor: 'leapmotion'},
4     sphero: {adaptor: 'sphero', port: '/dev/rfcomm0'}
5   },
6
7   devices: {
8     leapmotion: {driver: 'leapmotion', connection: 'leapmotion'},
9     sphero: {driver: 'sphero', connection: 'sphero'}
10   },
11
12   work: function(f){
13   }
14 }).start();
15  ```
16
17 At the moment, this code is not really doing anything but you can see how
       to specify which devices you are going to use. You have to specify a
       port for the Sphero because it connects to your computer via Bluetooth.
       To find the port for your own Sphero, run 'ls /dev/tty.Sphero*' in
       your console and replace the port in this code with the result you get.
18
19 The rest of the code goes inside the 'work' function as below:
```

work: function(my){ my.leapmotion.on('frame', function(frame){ if(frame.valid && frame.gestures.length > 0{ my.sphero.roll(70,0,1); }}} '''

如果 Leap Motion 控制器检测到任何种类型的手势，上面的代码将控制 Sphero 前进。当您在终端中搭建应用程序时，您可以执行以下操作：

- 将手指向上移动，使球停止并变蓝
- 向下移动你的手指/手，让球停止并变白
- 向前推，使 Sphero 向前直行（朝向 0° 方向）
- 向后拉，使 Sphero 向后（朝向 180° 方向）
- 向左滑动，使 Sphero 向左（朝向 270° 方向）
- 向右滑动，使 Sphero 右转（朝向 90° 方向）

就是这么简单!

原文链接：http://blog.leapmotion.com/controlling-sphero-leap-motion-cylon-js/

原文链接：https://www.wandianshenme.com/play/leap-motion-javascript-cylonjs-control-sphero