Android Spybot

ECE Capstone Project

Erik Bruckner - bajisci@eden.rutgers.edu
Jason Kelch - jkelch@eden.rutgers.edu
Sam Chang - schang2@eden.rutgers.edu

5/6/2014
# Table of Contents

**Introduction**........................................................................................................3
  Objective...........................................................................................................3

**Design**.............................................................................................................4
  Hardware...........................................................................................................5
  Software...........................................................................................................8
  Cost Analysis....................................................................................................10

**Future Work**..................................................................................................11

**Conclusion**.....................................................................................................11
Introduction:

With recent advances in consumer electronics, mobile phones have displayed an incredible range of possible applications. Given that unmanned surveillance systems are a rapidly growing industry, this project was interested in developing a system based around Android phones that could meet the basic requirements for a mobile surveillance system, while minimizing cost and user training. Through the development of this surveillance platform, the group members will be able to simultaneously gain experience with application development, robotics, and networking.

Objective:

The basis of this project is to develop an advance mobile surveillance platform that can quickly be deployed in the field with minimal cost and training. This is achieved using readily available commercial off the shelf parts for the chassis and drivetrain, as well as all electronic components. The user interface must be carefully designed to provide an immediate sense of familiarity that allows for easy use, even with minimal training. This objective is achieved using two cell phones, one mounted to the robot that sends video, and another to control the robot and receive the video. This will provide a simple, yet effective tool to enhance reconnaissance and security monitoring.
This diagram outlines how our various assets communicate with each other. The two android logos represent the two phones which talk to each other over WiFi and stream video. The main phone communicates to a bluetooth module which delivers the code to the Arduino Uno. The Uno decodes it, passes the information to the motor shield and controls the motors.
Hardware Design:

The SpyBot is designed to optimally provide mobility to the platform as a whole, as well as the mounted camera-phone. In order to meet those requirements, two brushed DC motors, and gears are attached to H-bridges. By using an H-bridge, the microcontroller can easily change the direction and speed of each motor independently, while running off a common power supply. The H-bridge is implemented in the arduino motor shield.

The bluetooth module is connected to the arduino board which communicate with each other through analog signals. The module receives its instructions from the android phone app. The H-bridge takes these signals and instructs the motors what to do. In addition, there is a servo mounted that can change the tilt angle of the mounted camera-phone.

The servo has a 470μf capacitor connected between the power and ground for stability. The servo has its own isolated power supply because it needs to constantly be powered to hold the arm in place. The servo has approximately 4.8kg/cm of torque, more than enough to support a cell phone.
The chassis is made with a combination of commercial, off the shelf parts, and custom designed parts. The double gear box has a gear ratio of 114.7:1 which provides a decent combination of power and speed for the robot.

The robot uses differential steering for easy usability and high maneuverability. Differential steering allows you to spin in place which takes away the need for a 360 camera rotation. This makes the design more compact and cuts the cost down, while still allowing the user to cover all angles. The motors operate at 6V and draw approximately 1mA each.
*Final robot prototype design including mounted second cell phone which handles video streaming function.*

*Robot during testing*
Software Design:

Overview:
The SpyBot components consist of two android mobile devices, an Arduino Uno, Arduino MotorShield and a Bluetooth module. The user’s android device is used as a Bluetooth controller, with a simple and intuitive GUI that anyone can pick up and use without prior training. In order to have the android application be as easy to use as possible, a traditional “tank treads” movement system is used, which is very intuitive to anyone with experience with RC cars. Several well-labeled options are also available, while limiting the amount of on-screen information that could confuse users. The user’s phone communicates with the mounted android device using Wi-Fi Direct to provide a streaming video feed from the robot’s point of view. The Bluetooth signals are interpreted by the Arduino Uno, which process the commands into mechanical motion via PWM signals sent to the Arduino MotorShield which drives the motors.

Arduino code:
The Arduino code was designed to interpret signals received through bluetooth from the phone controller. For each state that the robot can be in, a different code was assigned. A switch case is used, and for each different state, sends the proper signals to the motors and/or servo. For the servo to maintain its state, the servos current position is constantly sent to the servo. In addition, our code implements a Bluetooth disconnect fail safe, such that if it does not receive a command after a set period of time, it will trigger a flag which tells the robots motors to brake.

Android app design:
To fully implement the differential steering implemented by our tread based robot and simultaneously control the camera angles, we made use of two buttons for each tread and multitouch. Since android's views do not inherently incorporate it, an android view was programmed to overlay the UI in order to handle multiple touches. By maintaining coordinates and sizes of UI components, touches on the multi-touch layer were matched to its corresponding location on the screen to implement button presses.
Video Streaming code:

Video Streaming was accomplished by making use of Wifi Direct. Wifi Direct was used to maintain the goal of the robot being mobile and able to be used in any location/situation. It allows connections between two mobile devices without the availability of a router. When compared to bluetooth, Wifi Direct surpasses it in both speed and range with speeds of up to 250Mbps, which was a perfect fit for video streaming purposes. Instead of having a router as an access point to communicate between two devices, Wifi Direct allows a device to take the role of the access point, also known as the Group Owner. MJPEG was chosen due to its simplicity. MPEG, an alternate video encoding method, has a downside of suffering in quality during fast movement due to how its encoded. Since MJPEG is, essentially, independent JPEG images sent at a fast rate, it can handle fast movement and is our reason for choosing it. To transmit the MJPEG encoded video over a network we used the HTTP protocol over TCP. TCP allowed for less dropped frames and our high bandwidth from using WiFi Direct offsets TCP’s drawback of requiring to resend every dropped packet.

Here is the final GUI design, on the top left there are four different speeds the robot can operate at. There are two sets of Forward/Reverse buttons which enable differential steering and two buttons to control the
angle of the camera. Finally, in the center the video stream from the second phone is shown.

**Cost analysis:**

<table>
<thead>
<tr>
<th>Item</th>
<th>Quantity</th>
<th>Price</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>Arduino Uno Starter kit</td>
<td>1</td>
<td>$43.99</td>
<td>$43.99</td>
</tr>
<tr>
<td>Tamiya 70108 Tracked Vehicle Chassis</td>
<td>1</td>
<td>$20.17</td>
<td>$20.17</td>
</tr>
<tr>
<td>Arduino Motor Shield R3</td>
<td>1</td>
<td>$25.95</td>
<td>$25.95</td>
</tr>
<tr>
<td>Tamiya 70168 Double Gearbox L/R Independ 4-Speed</td>
<td>1</td>
<td>$12.63</td>
<td>$12.63</td>
</tr>
<tr>
<td>NEEWER® Aluminium Alloy Matte Pan and Tilt Bracket</td>
<td>1</td>
<td>$10.95</td>
<td>$10.95</td>
</tr>
<tr>
<td>4 “AA” Battery Holder</td>
<td>2</td>
<td>$2.29</td>
<td>$4.58</td>
</tr>
<tr>
<td>Wireless Bluetooth Transceiver Module</td>
<td>1</td>
<td>$10.99</td>
<td>$10.99</td>
</tr>
<tr>
<td>#6-32 x 3 IN Machine screw with nuts(4 pack)</td>
<td>2</td>
<td>$1.26</td>
<td>$2.52</td>
</tr>
<tr>
<td>Scotch Multi-Purpose Fasteners</td>
<td>1</td>
<td>$2.79</td>
<td>$2.79</td>
</tr>
<tr>
<td>NEEWER® Aluminium Alloy Matte Pan and Tilt Servo</td>
<td>1</td>
<td>$10.45</td>
<td>$10.45</td>
</tr>
<tr>
<td><strong>Total cost</strong></td>
<td></td>
<td></td>
<td>$160.01</td>
</tr>
</tbody>
</table>

The cost of designing the robot is fairly cheap. By purchasing the products in bulk and using old android phones the cost will be even cheaper. People constantly upgrade cell phones and leave the old ones sitting in a
drawer. They can easily be repurposed using our app and spybot. Combining all these factors will make a cheap commercialized product that can be marketed to special forces or individuals interested in security.

**Future Work:**

The future work of spybot can only be limited by your imagination. This highly customizable chassis was designed so that the user can easily add on parts. By adding an infrared or ultrasonic sensor an autonomous mode can be implemented and the robot will drive around collecting data. We can also run the app over wifi so it can connect to a website, this will allow security workers to watch multiple robots driving around a facility. The phone's microphone can be used to detect and transmit audio to alert the user. A picture capture mode could also be implemented so you can save footage. Another benefit of repurposing a phone as an IP camera, is one could make use of the broad range of features an average smartphone contains such as light sensor, GPS, compass and even a flash light. All these features could be implemented into our app.

**Conclusion:**

This project proved to be a challenging, yet rewarding experience allowing us to gain a breadth of knowledge in both designing and implementing a robot and android application. Though the prototype would need to be redesigned for mass use, it would certainly be a feasible and economical solution to anyone’s security and reconnaissance needs.