Landon Anderton, Alex Freshman, Kameron Sheffield, Sunny Trinh

SMART LOT
Intro

Engineer a system that will locate empty spots in a parking lot and report those empty spots to anxious parkers so they can find the closest parking spot available.
System Overview
TRENDnet Camera

- 640 X 480 at 30 frames a second
- Out of box wireless communication
- Runs an on board web server
- Can be set up to upload a still images to a server
Wireless Light

- Talks 802.11b/g with the server
- Receives data packets containing which parking lot regions are occupied as well as when to expect the next packet
- Sends ack messages
- Switches lights red or green
- Sleeps during down time
# Bill of Materials

![Images: sparkfun.com](https://via.placeholder.com/150)

<table>
<thead>
<tr>
<th>Product</th>
<th>Vendor</th>
<th>Quantity</th>
<th>Total Price</th>
</tr>
</thead>
<tbody>
<tr>
<td>TRENDnet - Wireless Internet Camera</td>
<td>newegg.com</td>
<td>1</td>
<td>$70</td>
</tr>
<tr>
<td>WiFly Shield 802.11b/g</td>
<td>sparkfun.com</td>
<td>1</td>
<td>$90</td>
</tr>
<tr>
<td>Arduino Main Board</td>
<td>sparkfun.com</td>
<td>1</td>
<td>$30</td>
</tr>
<tr>
<td>Arduino Stackable Header - 8 pin</td>
<td>sparkfun.com</td>
<td>2</td>
<td>$1</td>
</tr>
<tr>
<td>Arduino Stackable Header - 6 pin</td>
<td>sparkfun.com</td>
<td>2</td>
<td>$1</td>
</tr>
<tr>
<td>LEDs - Red</td>
<td>RadioShlock</td>
<td>32</td>
<td>~$5</td>
</tr>
<tr>
<td>LEDs - Green</td>
<td>RadioShlock</td>
<td>32</td>
<td>~$5</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td></td>
<td></td>
<td><strong>$202</strong></td>
</tr>
</tbody>
</table>
Hardware Interfaces

- Wireless Light to Server
  - 802.11g
  - Packet structure:
    - First 16 bits represent a light and whether to show green or red
    - Next 32 bits represent how long the light should wait before waking up in seconds

- WiFly Shield to Arduino
  - Connects directly to the Arduino using two 6 pin headers and two 8 pin headers
  - Arduino communicates with WiFly via an onboard **SC16IS750 SPI-UART bridge**
More Hardware Interfaces

- Wireless Camera to SmartLot
  - Wireless camera will upload images to a web server via FTP
  - SmartLot will check for new images and download them as they appear
Server Software Structure

- Networking
- Image Calibration
- Testing
- Occupancy Detection Algorithm
- Determines frequency at which to read images and send packets to the wireless light.
- Reads in new images at a given frequency.
- Builds the packets for sending to the wireless light.
- Receives acknowledgement packets from the wireless light.
- Uses a GUI to calibrate a given camera angle for parking spaces
- Writes text files containing coordinates for the calibrated angles
Test Gui
Uses a GUI to visually show the results of the Occupancy Detection Algorithm.

GUI allows the user to scroll between camera angles and see the results.

GUI shows red lines for spots found to be full and green lines for spots found to be empty.
3 Options to Try

- 1. Tonal Histogram
- 2. Color Histogram
- 3. Edge Detection Filter

Aforge.Net framework provides all the necessary tools.
Software Interfaces

- Calibration GUI to Occupancy Detection
  - Text file containing coordinates

```plaintext
calibration.txt
lotId: 0x0842;
numRegoins: x; // the number of regions in the lot e.g. rows
numSpaces_x: y; // the number of spaces per region
{(65:105-78:137), (78:137-122:139), (122:139-145:102)}
{(65:105-78:137), (78:137-122:139), (122:139-145:102)}
{(65:105-78:137), (78:137-122:139), (122:139-145:102)}
```
Software Interfaces

- Occupancy Detection to Test GUI
  - Hashmap: where the key is a region and the value is a list of open spaces for that region
Gantt Chart
Risks

- Getting the occupancy detection algorithm to work well in all conditions.
Questions?...