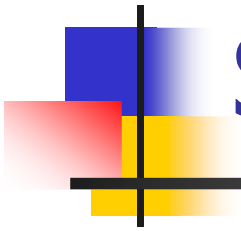
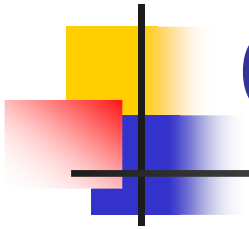


Hummerbot and the Speedway Competition

IEEE Robotics & Automation Meeting

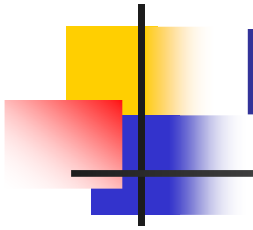
November 10, 2008

Robert Noteboom



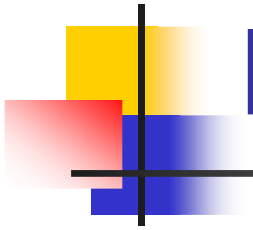
Overview

- Background
- Hummerbot Components
- Hummerbot Software Architecture
- Vision Processing Algorithm
- Raceway Simulation Program



Background

- Hummerbot origins
  - Group project to replicate DARPA Grand Challenge on a smaller scale
  - Manually drive vehicle to waypoints and then let it autonomously return
- Original Hummerbot Configuration
  - Modified RC Hummer from "Toys R Us" to accept standard RC PWM signals for steering (Parallax standard servo) and throttle (Electrifly C-20 ESC)
  - Interfaced BX-24 to a GPS receiver and the steering/throttle controls and added software



Background

- Hummerbot Speedway Competition Mods
  - Navigation Sensors: Replaced GPS receiver with a Webcam
  - Low-level Command Processing: Replaced BX-24 with an Arduino (sends steering and throttle commands)
  - Vision Processing: Added laptop PC with OpenCV to perform vision and command processing
- While other navigation sensors were considered, vision was considered critical to success in competition

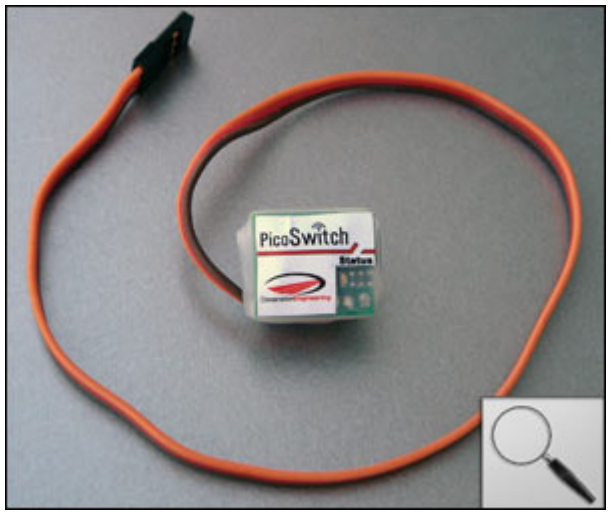
Components

Hummer mobile base
with Mabuchi RS-540RH/SH Motor

Parallax Standard Servo
for steering

Components

RC controller/receiver (JR Racing XR3i 3-Channel controller / receiver)



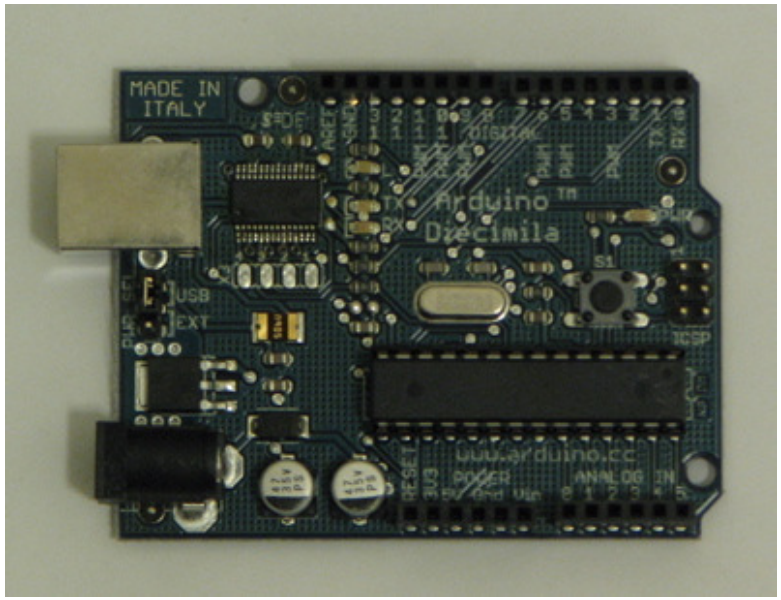
Radio Controlled Relay (Dimension Engineering PicoSwitch)

Components

Electronic Speed Controller
(Electrifly C-20 ESC)

Webcam (Logitech QuickCam
Communicate Deluxe)

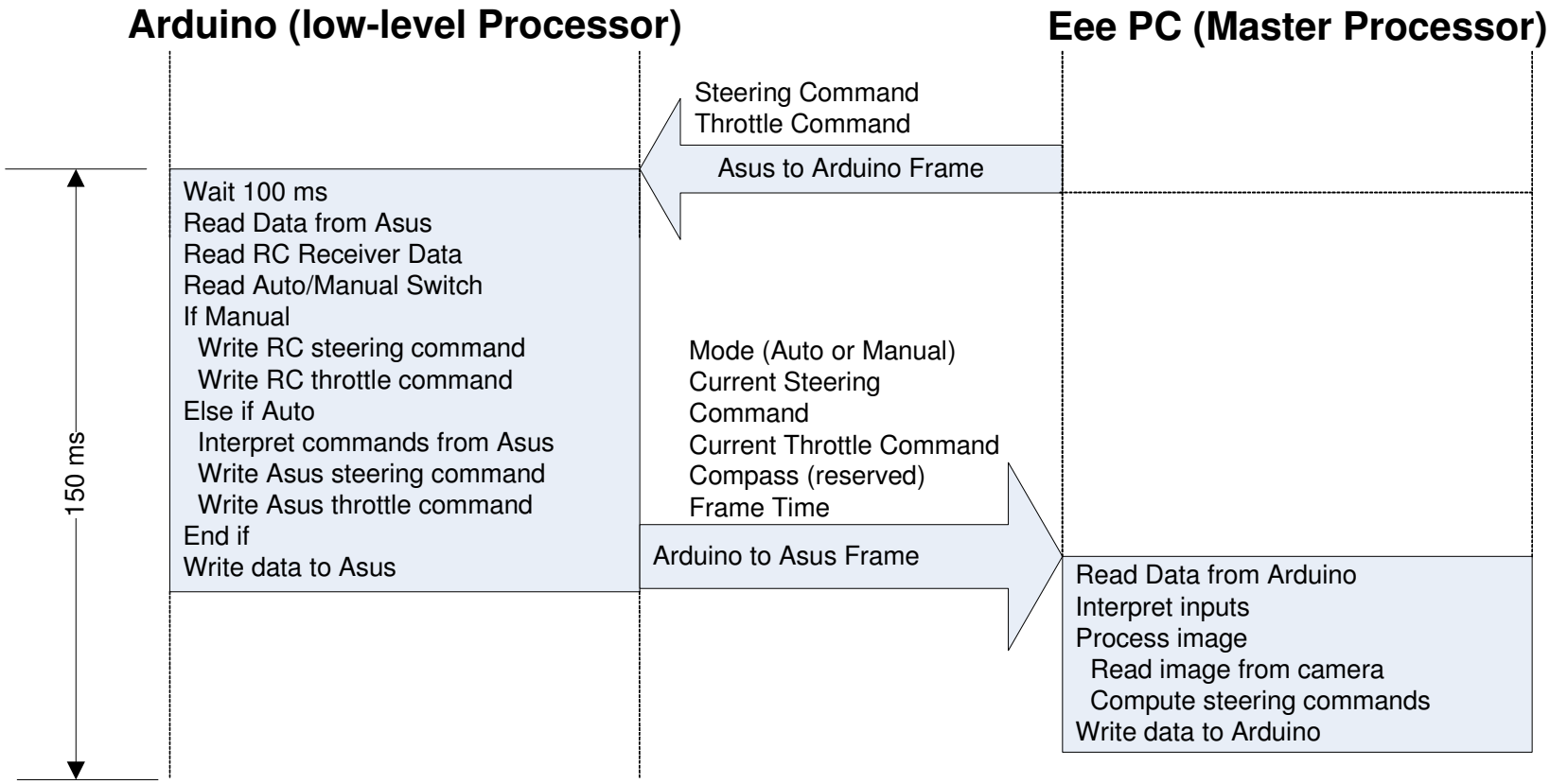
Components



Arduino (Diecimila)

Asus Eee PC 900

Software Architecture

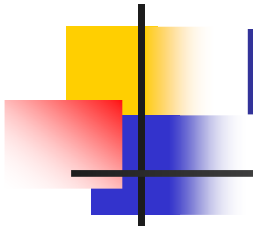


Vision Processing Algorithm

- Algorithm called once per 150 ms frame to process image and issue commands
- Open Source package (OpenCV) used extensively for image processing
- Consists of two parts
  - Capture image and find relevant components in image
  - Determine steering and throttle commands based upon location of components

Processing Image

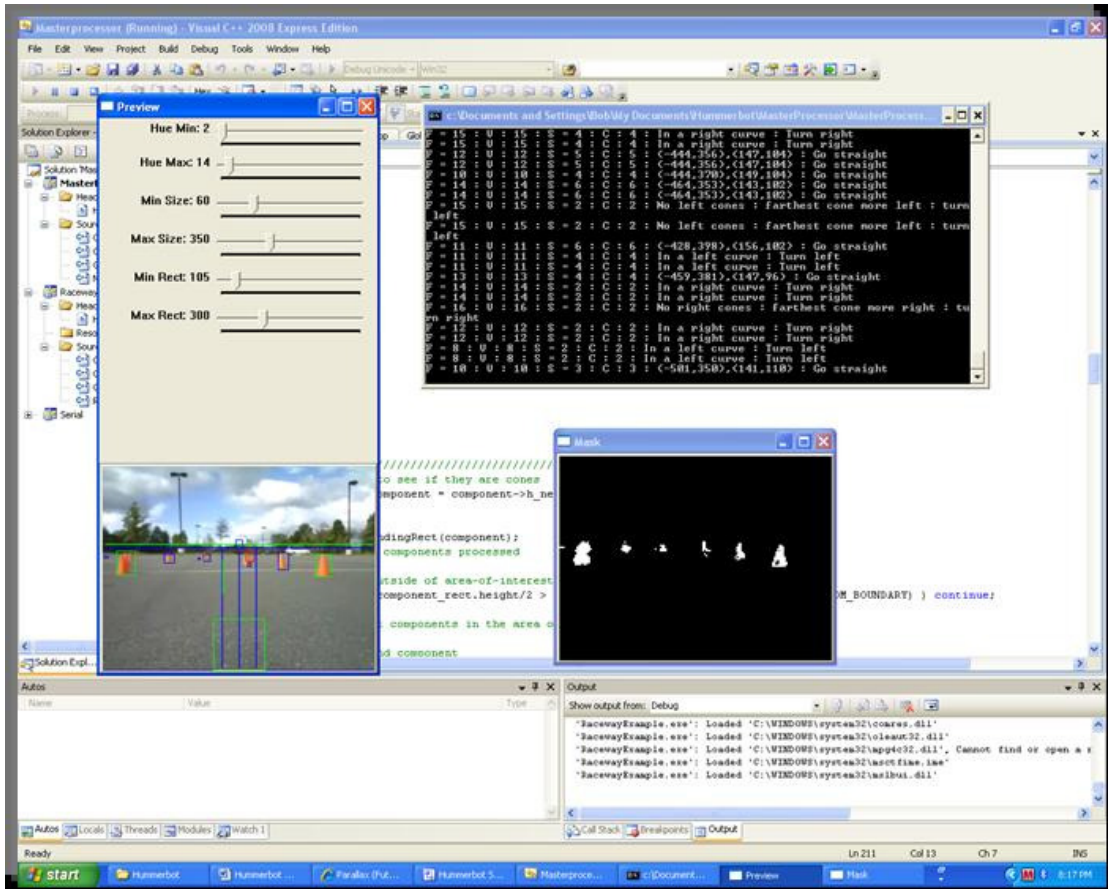
Capture Image from Webcam (cvQueryFrame)
Convert to HSV (cvCvtColor)
Get Hue Channel (cvSplit)
Threshold Hue Image (cvInRanges)
Find all connected components (cvFindContours)
For each component
{
    Determine bounding rectangle (cvBoundingRect)
    Ignore if outside area of interest
    Determine perspective size correction
    Ignore if too small, too large, or wrong shape
    Determine if component
    {
        Closest left cone
        Farthest left cone
        Closest right cone
        Farthest right cone
    }
}



Determining Commands

```
If inadequate number of cones found
        Lost: continue slowly with last command
Else if only right cones found
        Turn in direction of line from closest to farthest cone
Else if only left cones found
        Turn in direction of line from closest to farthest cone
Else if closest left cone is farther than farthest right cone
        Turn left
Else if closest right cone is farther than farthest left cone
        Turn right
Else use location of midpoint between closest left and closest
right cone to determine steering command
{
        Perspective correction for closest left cone
        Perspective correction for closest right cone
        Determine midpoint between cones
        If midpoint is to right
                Steer right
        Else if midpoint is to left
                Steer left
        Else midpoint is near middle
                Go straight
}
```

Raceway Simulation Program



Raceway Simulation was used to analyze vision processing and steering commands

- **Examine frame-by-frame behavior**
- **Adjust object recognition parameters**
- **Display resulting commands**

For More Information

- Source code and test videos can be found at:
  http://code.google.com/p/hummerbot/downloads/list
  - MasterProcessor - 2008_09_28.zip (Asus code)
  - Hummerbot2.zip (Arduino code)
  - A number of test videos
- Asus code built using Microsoft Visual C++ 2008 Express Edition with OpenCV (Open Source package)
- Contact me at: bob@noteboom.org

Special Thanks to…

- All the members of the Hummerbot team including Shivang Patel, Jaskarian Jamwal, Dan van Niekerk, Brijseh Sirpatil, and Marius

- Everyone at the IEEE Robotics and Automation Society and University of Maryland that helped with the Speedway Competition!