Design and Realization of an Intelligent Unmanned Ground Vehicle

Justin Barrett, Rob Fitzpatrick, Chris Gamache, Ricardo Madera, Adam Panzica, Benjamin Roy, Daniel Sacco, ViktorasTruchanovicius, Bohua Wang

WPI

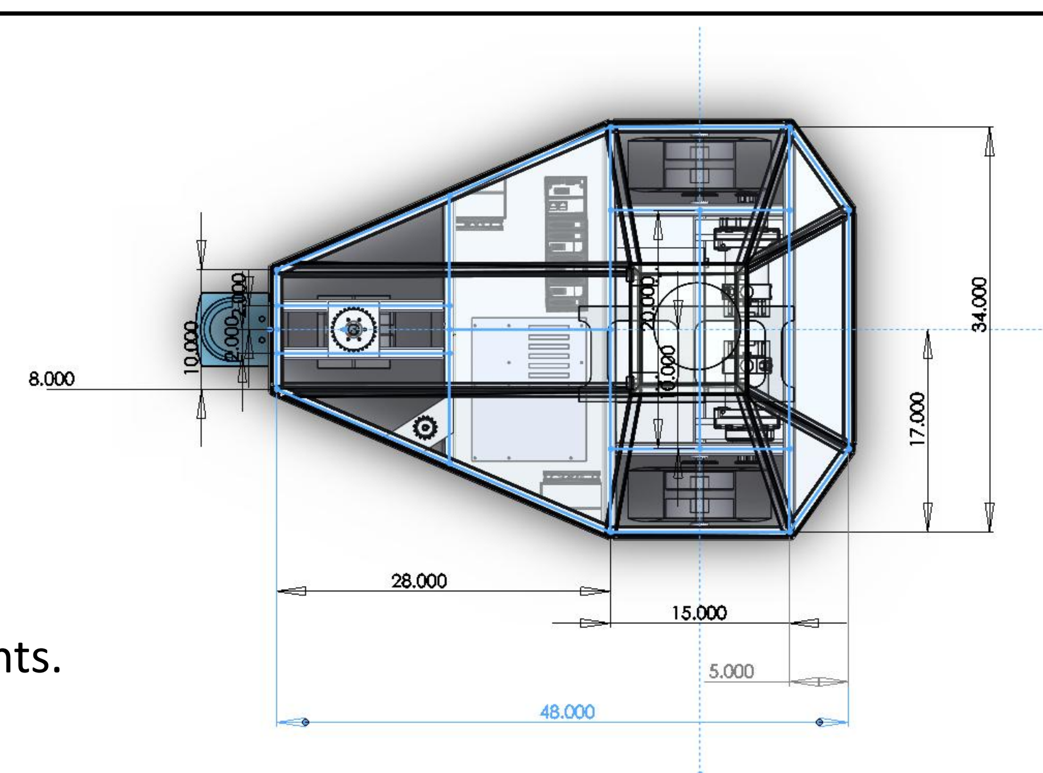
Prometheus
Worcester Polytechnic Institute

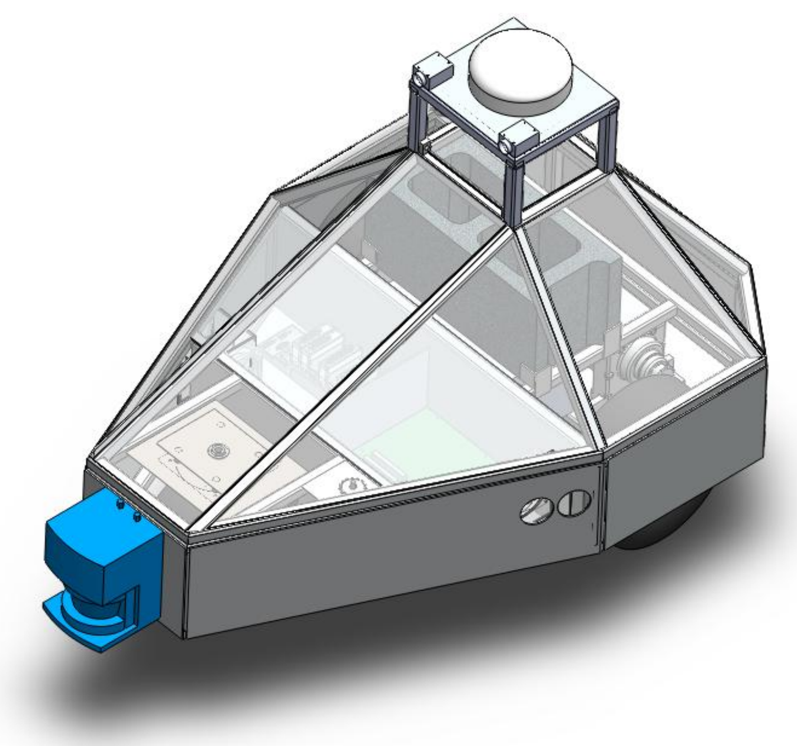
Advisors:
TaskinPadir, Michael Ciaraldi,
Gregory Fisher, Kenneth Stafford,
William Michalson

This project focuses on designing a computational framework and developing software for an intelligent unmanned ground vehicle (UGV) which will become WPI's first entry to the Intelligent Ground Vehicle Competition (IGVC). IGVC challenges students to design and program a fully autonomous UGV that can locate and avoid obstacles, stay within the boundaries of a lane, navigate to GPS waypoints and implement a communications system using the Joint Architecture for Unmanned Systems (JAUS) protocol.

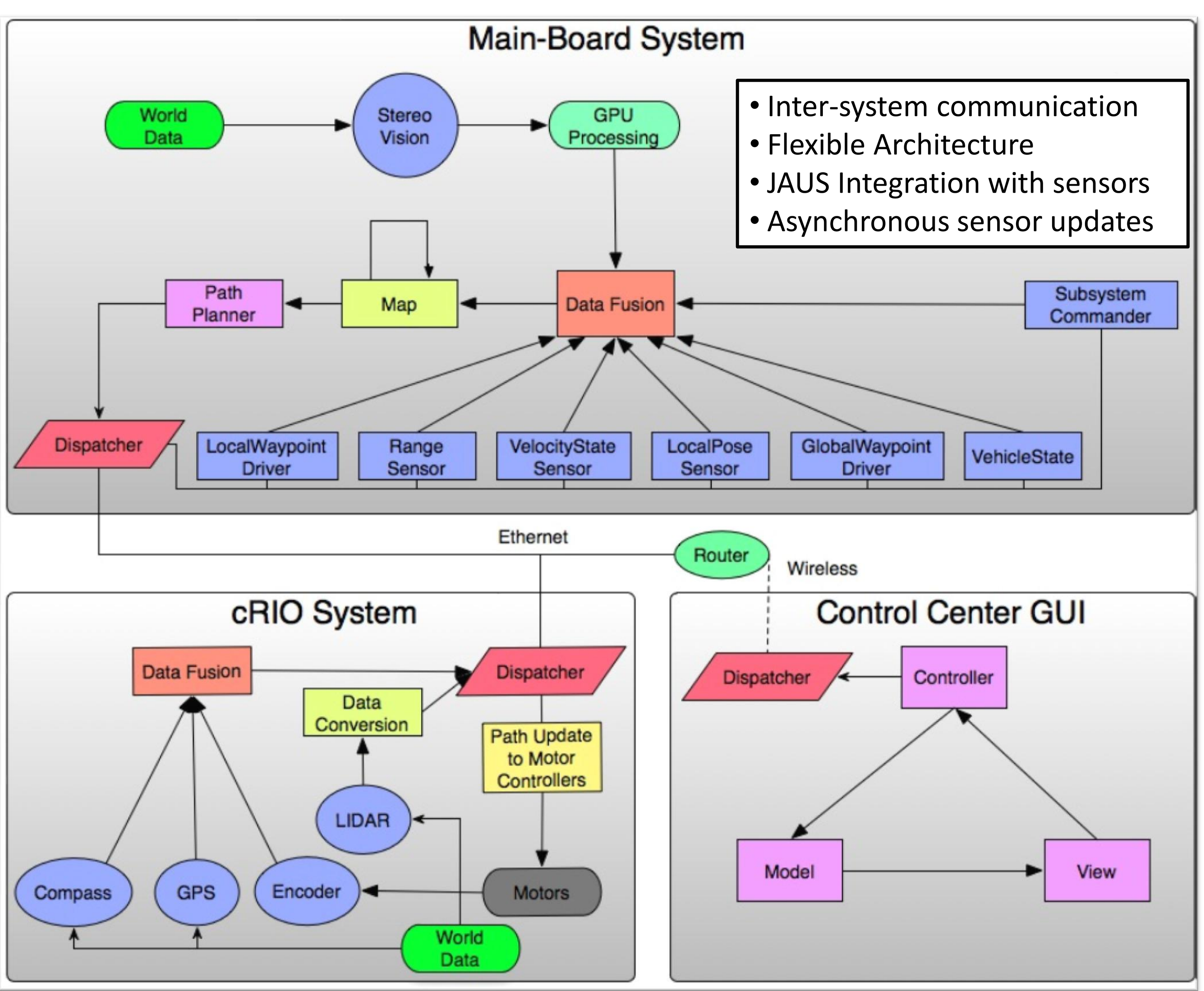
Design Features

- Rear differential drive using two NPC-T64 DC motors.
- One steered front wheel.
- Custom water hardened aluminum chassis.
- Two led acid batteries.
- National Instruments cRIO Controller
- Main board computer using NVIDIA's Tesla 1060c GPGPU
- Path calculation using "Driving with tentacles" approach.
- Use of OpenJaus library to implement JAUS challenge requirements.



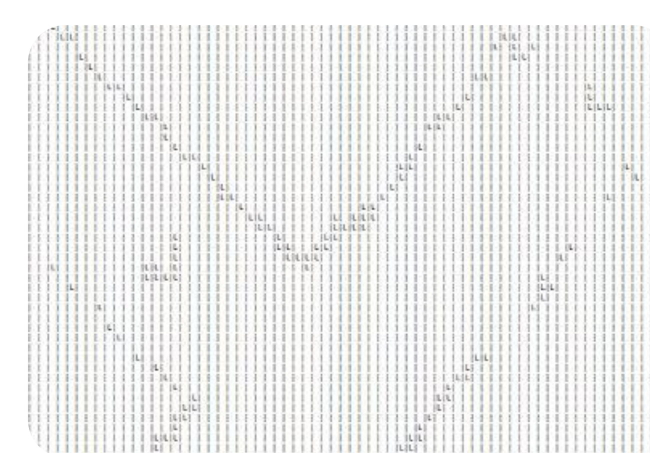


System Architecture Overview



Mapping and Localization

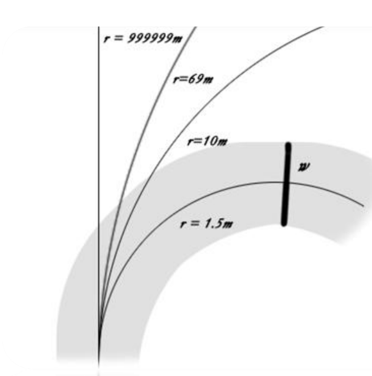
- Map type: local ego-centered
- Map is stored as 2D array of grid cells.
- A cell stores probability and object type.
- Map is updated asynchronously with each LIDAR or Stereo Vision data update.
- Probabilities of cells are recalculated using the Gaussian distribution



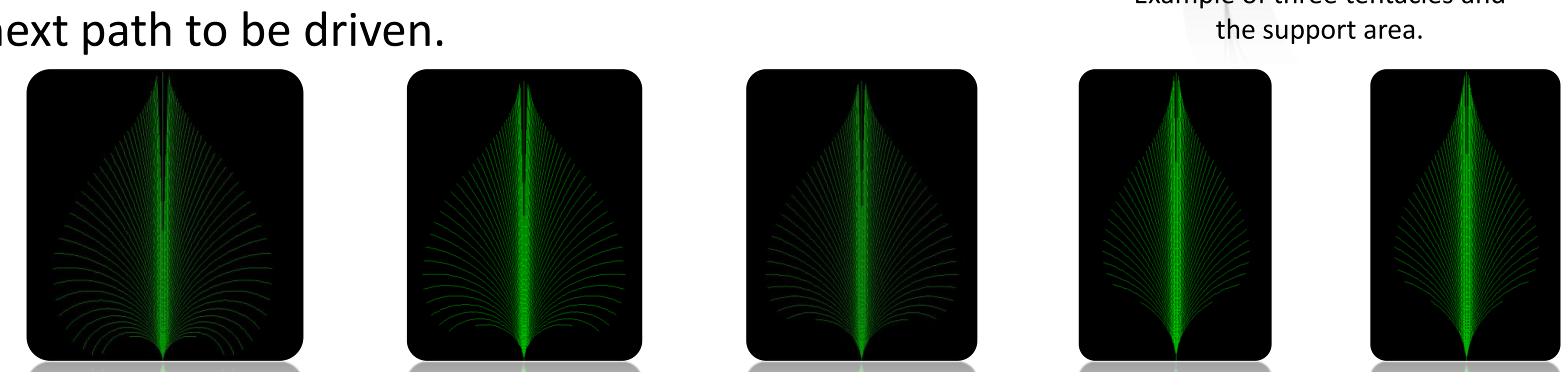
Example of the map being printed in ASCII

Path Planning

- 5 Speed sets (0.4m/s to 2.2m/s).
- 81 Tentacles in each speed set.
- Longest tentacle: 9.5m.
- Shortest tentacle: 1.5m.
- Segmented into bins that have weights assigned.
- Tentacle with the lowest weight is picked as the next path to be driven.



Example of three tentacles and the support area.



Speed Set 1 | Speed Set 2 | Speed Set 3 | Speed Set 4 | Speed Set 5

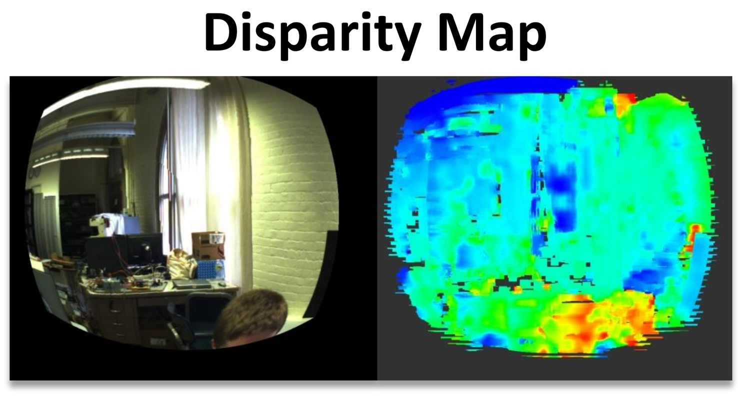
Stereo Vision

Rectification converts two images into the same image plane. Segmentation groups pixels based on color and is later used to refine the image disparity map. Distances to objects are determined using calculated disparities.

Rectification

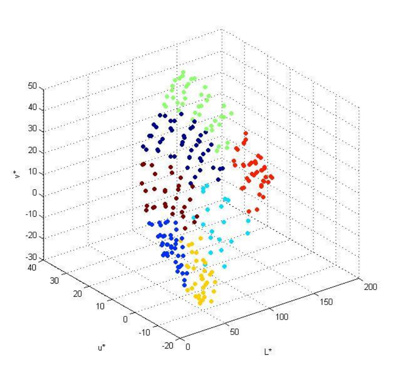
Rectified images

Disparity Map

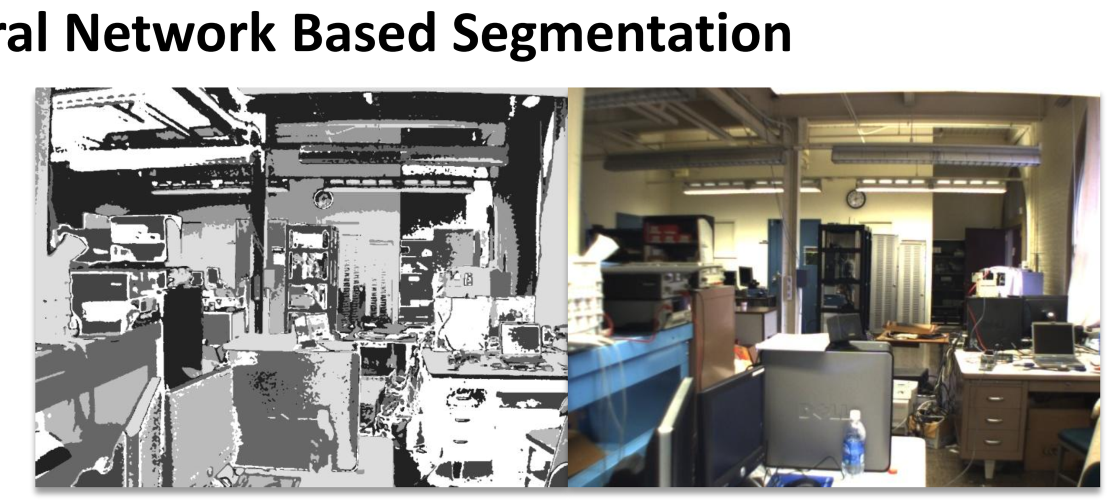


Rectified Image | Disparity map

Neural Network Based Segmentation



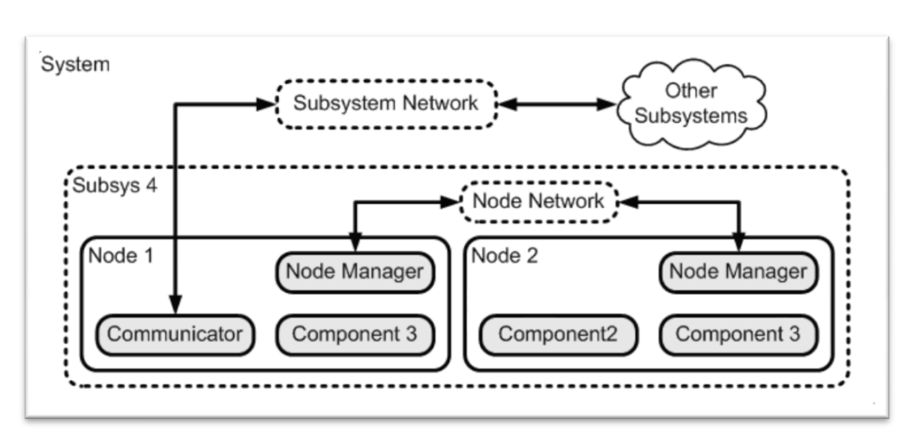
Clustered neural network used to segment the images.



Segmented image | Original RGB image

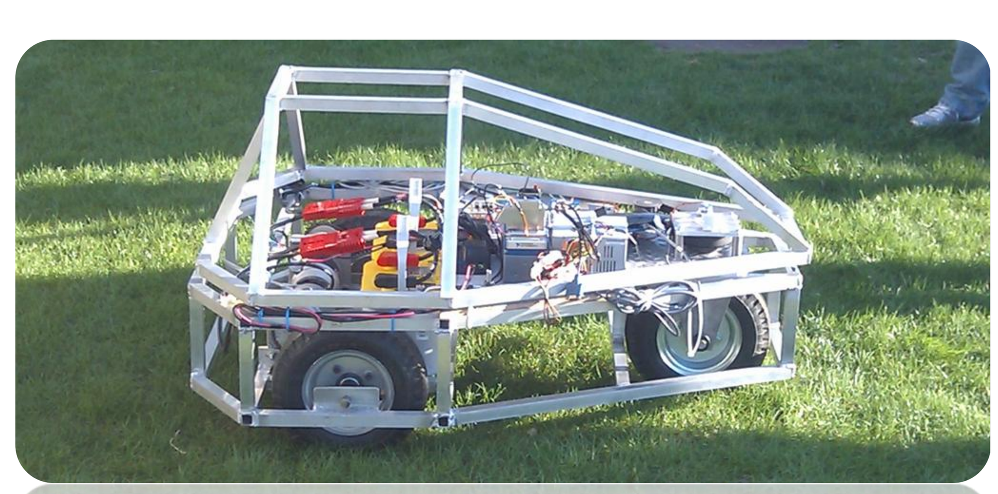
JAUS – Joint Architecture For Unmanned Systems Challenge

- Responding to discovery, management and position queries and reports.
- Implemented using OpenJaus, open source implementation of JAUS Reference Architecture

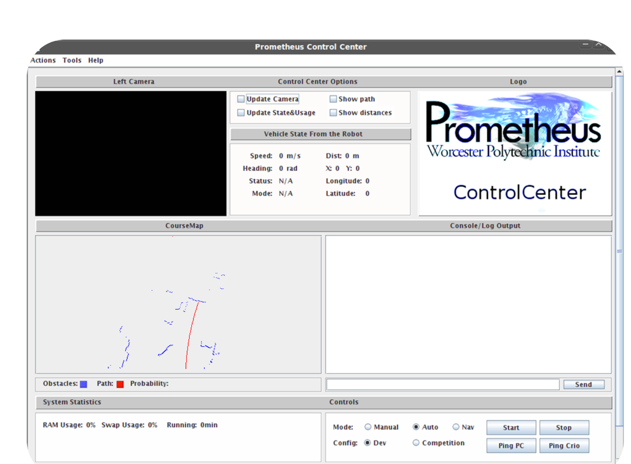


http://www.openjaus.com/media/Final%20RA%20V3-3%20Part%201.pdf

Results



Prometheus



Example of the Control Center plotting LIDAR data and the best path