PROJECT PLAN

Intelligent Ground Vehicle Competition

Florida Tech Senior Team Members

Name	Major	Email
(Team Lead) William Nyffenegger	CSE Spring 2017	wnyffenegger2013@my.fit.edu
Adam Hill	CSE Spring 2017	ahill2013@my.fit.edu
Brent Allard	CSE Spring 2017	ballard2014@my.fit.edu
Chris Kocsis	CSE Spring 2017	ckocsis2007@my.fit.edu

SEPTEMBER 7, 2016

FLORIDA INSTITUTE OF TECHNOLOGY
Florida Tech IGVC Team

Goals

The goal of Florida Tech IGVC is simple: win the annual Intelligent Ground Vehicle Competition (IGVC) in Michigan.

IGVC is an autonomous robot competition with random courses including random obstacles, random paths, and random goals. There are two courses ranging from a quarter to half a mile long; the robot must be entirely autonomous for the duration of the course.

Florida Tech IGVC (FIT) is collaborating with Florida State University IGVC (FSU) to produce this robot. FSU will design and produce the chassis of the robot, software for motor control, and integrate sensors into the robot; FIT will produce the software, configure sensors, and begin testing the robot. The teams will ten coordinate the integration and testing of the complete system.

The long term goals of the project are broader. The goal of the project is to consistently compete at the annual Intelligent Ground Vehicle Competition in Michigan for the next five years. Additionally, the project must continue developing a relationship with FSU to hopefully create opportunities for new collaboration projects in the future. With those two goals in mind, the robot and software developed need to be extendable. Another goal of the project is to use advanced sensors and software to produce a state of the art vehicle and software development process.

To accomplish these goals, the team must develop a robot and software capable of autonomous course appraisal and navigation including these core capabilities:

- Intelligent navigation and planning
- Image processing and computer vision
- Motor control
- Precise and accurate position estimation
- Accurate and repeatable course simulation
- Multi-language interaction
- Hardware and software integration
- An extendable software framework

Key Software Features

The major goals concerning developing the robot correspond almost exactly to the key features of the software which we will develop.

These key features are:

- Object recognition based on depth and color data
- Position estimation based on inertial and GPS calculations
- Intelligent navigation and planning
- An extendable software framework
- An adaptable modular software system

The first three features correspond to developing a representation of any given course. The remaining features focus on the long term goals of the project.

Novel Features

IGVC began twenty-five years ago as a remote controlled competition and has since shifted towards autonomous navigation. The course itself is extremely challenging.

The course features:

- An unknown, random map which cannot be previously mapped
- Obstacles and roads, which require object recognition and motion planning
- Software memory of a course
- Variable conditions for sensors
- Limited speed and capabilities

These features require us to consider the following innovations:

- Real time map building of an unknown course
- Navigating a limited space in real time
- Position triangulation and estimation accurate within a foot
- A novel robotics communication framework using Advanced Message Queueing Protocols
- Experimental sensors and computing hardware with extraordinary capabilities
- Collision detection optimization using GPU programming
- Obstacle detection and image recognition in real time

The problem of building a course in real time is an ongoing problem for autonomous vehicles in general. Real time map building is inhibited by sensor accuracy and precision limitations as well as limited hardware capabilities. Both of these technical challenges require advanced approaches, particularly position estimation. Furthermore, navigating constructed spaces in real time is exceedingly difficult.

The experimental hardware chosen is cutting edge with increased power for image processing and computing in general. The devices are meant for drones and autonomous vehicles—including potentially cars. Several of the experimental sensors give unprecedented accuracy and precision especially in depth and color sensing. Additionally, the vision aspect of the project includes obstacle detection and recognition, which must also occur in real time. The GPU will be relied upon to handle significant portions of the image processing to increase the speed of calculations relating to position, object recognition, and distance from objects for navigation.

Running navigation, map building, motion planning, and image processing in real time requires parallelization. An advanced, standardized messaging framework, which is not normally used in robotics, will allow us to make each major piece of software independent. The framework not only allows independence; but, naturally precludes the entire piece of software from deadlocking.

Technical Challenges

The major technical challenges associated with the robot focus on the difficulty of producing:

- Accurate simulations of a course environment
- A map-building algorithm that builds and updates a course representation in real time
- Parallelized modular systems for image processing, motor control, and navigation
- Image processing and obstacle detection in real time
- Integrating high level software with low level sensors
- Communication between independent threads running in multiple languages
- Accurate position estimation

These challenges manifest themselves not in designing a basic Artificial Intelligence; but, in building a representation of the course which that AI may use. Furthermore, the inaccuracies in estimating position propagate. Our position estimation must be extremely accurate otherwise our representation will not be accurate.

Parallelism will be a key challenge. We will have many threads running to complete image processing, calculate the course map, run the artificial intelligence, and control the motors. All of these threads must communicate; but, these threads also need to be independent. The robot as a system must not deadlock if a single thread fails. Instead, it must adjust for that failure and recover that thread. These threads will also be running in multiple languages and need to communicate. To facilitate communication, we will use JSON and RabbitMQ to design a language independent messaging framework.

Another major challenge is integrating serial communications to sensors with high level interfaces. These sensors must be temporally synchronized. Meaning a position estimation (involving a GPS location and IMU data) must be made at close to the exact time a frame is taken otherwise the frame itself is meaningless.

Fall Milestones

0.1 Finished Simulation, prototyped AI, and prototyped robot

- Decide all sensor configurations
  - Determine whether current equipment is sufficient
- Design overall software structure
- Finalize mapping and navigation strategies
- Determine how to maximize GPU usage for computer vision
- Prototyped GUI and robot main controller
- Purchase position estimation sensors as required
  - Determine whether current equipment is sufficient
- Finished simulation and communication framework
- Create test plan for developing image processing
- Create requirement document
- Create metrics for measuring AI performance

Task	Will	Adam	Chris	Brent
Sensor Configurations	20 %	20%	40%	20%
Software Structure	25%	25%	25%	25%
Mapping and Navigation	20%	30%	20%	30%
Maximizing GPU	10%	10%	70%	10%
Prototype GUI & Main	40%	10%	10%	40%
Simulation & Communication	70%	10%	10%	10%
Position Estimation Sensors	30%	30%	20%	20%
Create Requirements Doc.	40%	20%	20%	20%
Test Plan Doc.	10%	20%	35%	35%
Create Design Doc.	20%	20%	20%	40%

0.2 Basic line detection and obstacle recognition

- Finished GUI and fully structured main controller
- Prototype AI and map building algorithms being developed on simulations.
- Prototype motion planning software
- Prototype position estimation software based off purchased sensor units
- Working frame grab and prototyped transformations for computer vision
- Finished requirements document

Task	Will	Adam	Chris	Brent
Finished GUI	30 %	30%	10%	30%
Prototype AI & Map Building	10%	40%	10%	40%
Prototype motion planning	30%	30%	10%	30%

Sensor Position Estimation	20%	30%	30%	20%
Frame Grab & Proto. Transforms	20%	-	80%	-
Finished Requirements Doc.	40%	20%	20%	20%

0.3 Completed AI, integrated sensors, and running software suite

- Constructed robot with sensors tested on the robot
- Complete map building algorithm with interface to computer vision
- Functioning simulation using realistic data
- Complete main controller with communications interface for all units
- Detailed testing plan for validating performance
- Finished motion planning software

Task	Will	Adam	Chris	Brent
Constructed Robot	40 %	20%	20%	20%
Map Building & Interface	10%	30%	30%	30%
Functioning Simulation	20%	40%	20%	20%
Main Controller & Comm. Int.	40%	20%	20%	20%
Detailed Testing Plan	20%	20%	30%	30%
Motion Planning	25%	25%	25%	25%