Faculty Advisor Statement

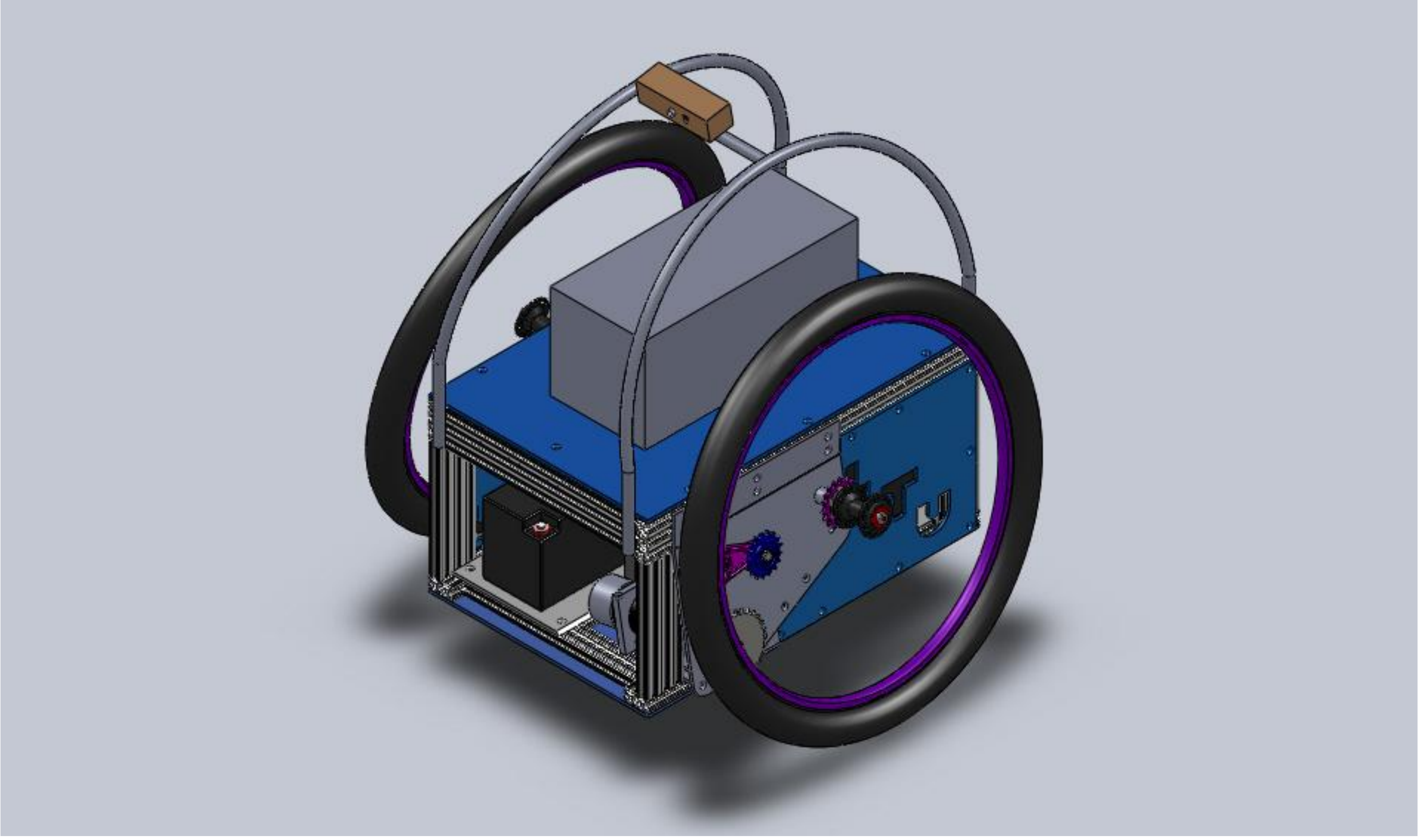
I, Dr. CJ Chung of the Department of Math and Computer Science at Lawrence Technological University, certify that the design and development on the VU**LTU**RE platform by the individuals on the design team is significant and is either for-credit or equivalent to what might be awarded credit in a senior design course.

Signature Date

The VU**LTU**RE platform is a new design from the ground up. Subsystems that are significantly different from those used in previous years include:

- Mechanical design, including
    - Chassis
    - Suspension
    - Drive train
- Electrical design, including
    - Voltage conversion system
    - Computer systems
    - E-Stop wireless sensor and interface circuitry
    - GPS sensor
    - Camera sensors
- Software design, including
    - Vision processing
    - Path planning
    - User interface
    - JAUS

VU**LTU**RE



Design Report

for the

2011 Intelligent Ground Vehicle Competition

prepared by

Lawrence Technological University

VU**LTU**RE Design Team

1. Introduction

This report describes the VU**LTU**RE robotic platform, designed and built for the Intelligent Ground Vehicle Competition (IGVC). This report is organized into sections describing the design team, the design processes, and various aspects of the design, followed by performance and cost information.

2. Team Organization

The VU**LTU**RE design team is comprised of the following members:

Name	Course of Study	Role(s)
Dr. CJ Chung	(Professor of CS)	Faculty Advisor
Brace Stout	MSCS	Team Captain
Daniel Anderegg	BSCS	Software
Chris Anders	BSCS	Mechanical
Jamie MacLennan	BSME	Mechanical
Ryan Matthews	BSCS	Software
Dan McGee	BSCS	Software
Johnathan Nabozny	BSCS	Software
Jonathan Ruszala	MSCS	Software, Electrical
Sato Taiga	MSCS	Software
Paul Wright	BSTM	Mechanical

3. Concept Development

A post-competition review of the 2010 IGVC results identified the areas in which our CU**LTU**RE SHOCK entry distinguished itself, as well as those areas in which improvement was needed. When the VU**LTU**RE team was formed, a number of 'brainstorming' sessions were held in which each of the problem areas was considered and possible solutions discussed. Our strength areas were also examined for opportunities for improvement. The results of these discussions and analyses are the basis for the innovative features found in the platform.

3.1 Failure Analysis

The following areas were identified as needing improvement during failure analysis. These received special attention in the design of the platform:

- **Wireless E-Stop** – The wireless E-Stop system seemed to have intermittent problems with range and sensitivity. This was traced back to the antenna being used in an attempt to extend the range of the system. Analysis showed that the antenna was poorly matched to the frequency being used by the wireless E-stop transmitter.
- **Suspension** – Large forces caused deformations of components of the stabilization system. While no catastrophic system failures were attributed to this, a more robust design was needed.
- **Vision Processing Software** – Our vision processing software was unable to clearly distinguish the white markings on the grass of the field, particularly in direct sunlight.
- **Accessibility** – The design of the battery compartment left the batteries captive, and to a large part, inaccessible.
- **Electrical Power** – We ran low on power for the non-propulsion electrical subsystems on a number of occasions, despite continuous charging of batteries during our downtime.
- **Battery Charging System** – A malfunction in the battery charger led to overheating of one of our batteries, resulting in decreased capacity.
- **Clearance** – Our platform experienced difficulties in maneuvering over “speed bumps”. This was traced back to an inadequate ground clearance.
- **GPS Performance** – Our platform was plagued with extended periods of time during which no GPS information was available. No specific cause was identified, and so the failures were attributed to the GPS system being used.
- **Drive Train Backlash** – The considerable forces involved in the acceleration of the platform caused some drive components to slip in addition to stretching the chain slightly. This led to the chain becoming somewhat loose, which further exacerbated the problem. As the encoders are mounted on the motor shaft, dead reckoning calculations were off as a result.

3.2 Identified Strengths

The following areas were considered strengths of our previous entry, but were examined for potential refinements:

- **Symmetric, Low-Moment Design** – The stabilized two-wheel design has proven to be superior in many aspects, including stability and maneuverability.
- **Large Drive Wheels** – The large drive wheels provide isolation of the platform from shock, and provide a larger contact patch than similar entries with smaller wheels.
- **Low Center of Gravity** – A low center of gravity provides stability during maneuvering and stopping. Furthermore, having the center of gravity below the wheel axles makes our platform inherently stable, as the low-energy state is the normal upright position.
- **Vision-Based Obstacle Detection** – The stereo cameras used for obstacle detection are deemed superior when evaluated with other technologies in the areas of size, weight, power, and cost (SwaP-C). Due to their solid-state construction, they also have superior performance with regards to mean time between failures.
- **DC-DC Electrical Conversion** – Using direct DC-DC conversion for the electrical systems decreases complexity, parts count, and wasted energy (as opposed to the use of a DC-AC converter coupled with AC-DC power ‘adapters’).

3.3 VULTURE Design Goals

The VU**LTU**RE platform represents an evolution of the CU**LTU**RE SHOCK platform, with improvements and refinements to nearly every subsystem, as well as new features to accommodate 2011 IGVC rule changes. The following summarizes the design goals for the platform, and highlights the innovations and improvements that these goals represent over previous years' entries.

- **Compact Size –** By minimizing the footprint of the platform, there is more room to maneuver within the competition area. A hard limit of 29.5 inches is imposed on the width, to ensure that the platform can be maneuvered easily and without modification through standard-sized doorways.
- **Large-Diameter Wheels –** The use of large-diameter wheels allows the center of gravity to be positioned below the wheel axles, which greatly enhances stability. The relatively large volume of air contained within the tires provides some measure of shock isolation. Finally, the large-diameter wheels provide a larger contact patch than similar wheels of smaller diameter.
- **Wide Stance** – The stance of the platform is maximized within the width constraints by angling the wheels. This provides greater stability on turns and inclines. This represents an innovative change to the design over previous years' entries.
- **Low Center of Gravity** – The center of gravity of the platform is placed as low as possible while maintaining minimum ground clearance. This lends to platform stability during maneuvers and while on inclines.
- **Low Moment of Inertia** – Placing dense components close to the vertical axis enhances maneuverability, as less force is required to change platform direction.
- **Backlash Management** – Backlash is minimized through the introduction of chain tensioners in the drive train. These maintain the tension in the chain and absorb sudden changes in force, reducing wear and tear on chassis and drive train components while aiding in accurate dead reckoning. This represents an innovative change to the design over previous years' entries.
- **Swappable Batteries** – Incorporate swappable battery banks into the design, allowing banks to be charged while platform is in use. All battery banks should be identical.
- **Vision-Based Obstacle Detection** – Use stereo cameras to sense the immediate environment. Develop and implements vision-processing software to identify and categorize obstacles.
- **Ease of Maintenance** – Minimize the use of custom-designed and custom-built components to aid in ease of maintenance. Where custom components are necessary, make and keep spares on hand.
- **Prefer Solid-State Components** – Where possible, use solid-state components in preference to components with motors. As the platform is mobile and can potentially be subject to shock, solid-state components will provide greater reliability. For example, choose cameras over a laser measurement device, and use solid state hard drives in the computational components.
- **Detect Battery Voltage Levels** – Incorporate voltage sensors to detect low-battery conditions. This represents an innovative change to the design over previous years' entries.
- **Improved E-stop** – Simplify the E-Stop electronics, and improve E-Stop reliability.

- **Increased Top Speed** – 2011 IGVC rules call for a top speed of 10 mph. The platform should be able to quickly reach and maintain this speed over level ground, as well as accommodate 15% gradients at a minimum of 8 mph.
- **JAUS Conformance** – Implement the JAUS features described in the 2011 IGVC rules document.

4. VULTURE Design

Platform design was divided into three main categories, or groups: mechanical, electrical, and software. Individuals were assigned to one or more of these groups based on their experience and areas of expertise. Beginning in December 2010, twice-weekly meetings were held to track status, as well as aid in collaboration and information exchange between groups. Each individual was responsible for tracking hours worked. The following sections detail the development of the various design aspects.

4.1 Mechanical Design & Fabrication

4.1.1 CAD & Modeling

SolidWorks® was used as the basis for capturing the mechanical design of the platform. This selection was based on the team members' experience with the tool, and its availability to students. An Internet "drop box" was used to share files amongst the group during development. During fabrication, 2D prints and dimensions exported from SolidWorks® were used for shaping components. Figure 1 highlights aspects of the mechanical design.

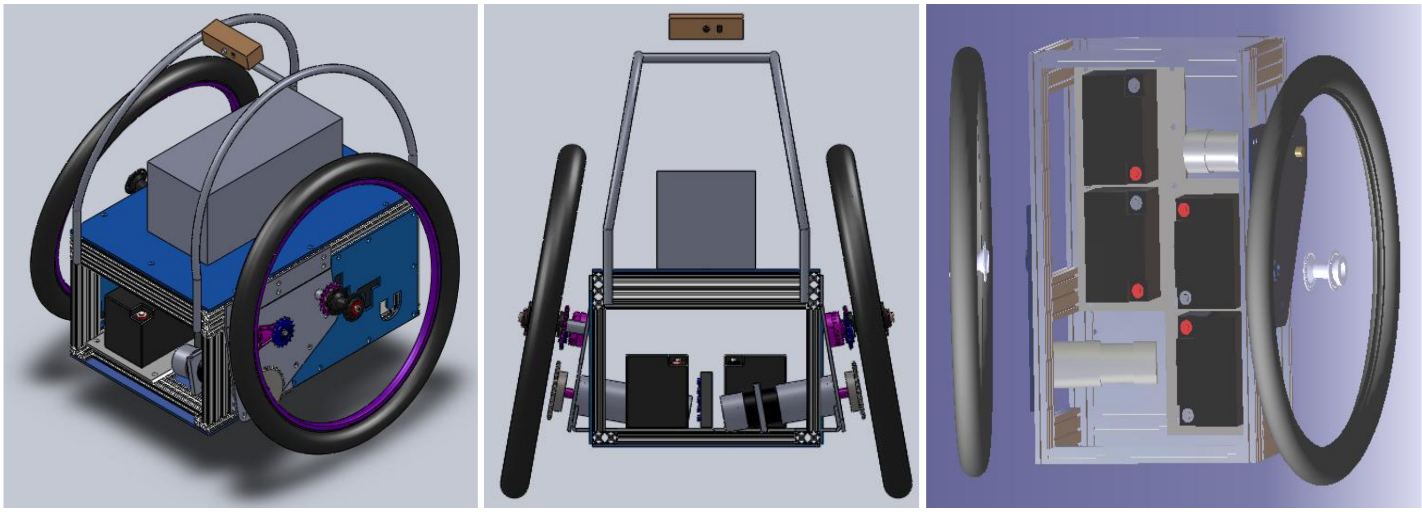


Figure 1 – Mechanical Design Highlights

4.1.2 Drive Train

The largest commonly available bicycle-style wheels were chosen. These have roughly a twenty-nine inch diameter including tire. Analysis of drive requirements led to the selection of the same motors as used in recent years, as these motors have sufficient torque to handle the increased speed requirement. Matching the 10 mph speed requirement to the motor's RPM at peak efficiency provided the required gear ratio. Trading off size, weight, and chain stresses, we determined that a 28 / 18 tooth chain drive configuration using ANSI #40 chain best meets all constraints.

Chain tensioners were added to the design to dampen sudden changes in acceleration and to reduce backlash.

Mounting the wheels at an angle presents a number of challenges, one of which is power transfer from the motor to the drive sprocket. A number of alternatives were considered, and ultimately it was decided to mount the drive sprocket directly onto the motor shaft, and to mount the motor at an angle to the main body of the platform chassis. The standard-size keyway on the readily available 28-tooth drive sprocket was broached to match the metric keyway on the motor shaft, avoiding the expense of custom-made sprockets.

4.1.3 Wheel Mounting

SolidWorks® was used to evaluate various wheel angles; an angle of 10° was found to strike a good balance between platform width, ground clearance, and low center of gravity. Single-side mounting is used for the wheels, providing a wide stance, improved maintainability, and a cleaner look. A 36-hole pattern was selected for the wheels (as opposed to a 32-hole pattern) to provide maximum strength. A survey of available bicycle hubs led to the selection of a robust, machinable hub with an integrated flange (normally used as a disc brake mount). SolidWorks® was used to verify that we could easily mate this flange to the required 18-tooth sprocket. The hub was machined to accept sealed bearings that fit snugly over a ½” steel shaft that is fixed to the platform chassis via an aluminum mounting boss welded to the mounting plate. A socket head shoulder bolt secures the wheel hub to this shaft.

A mounting plate was designed that 1) provides the 10° angle between chassis and drive train, and 2) provides a robust structural component for mounting motors, chain tensioners, and wheel mounts. This plate is made from ¼” 6061-T6 aluminum, selected for its toughness, machinability, and welding characteristics.

4.1.4 Chassis

SolidWorks® was used to model a number of possible battery configurations. The configuration selected supports a swappable battery module concept while keeping the movement about the vertical axis to a minimum. Battery modules are identical; allowing one or more “hot” spares to be charged while the platform is being used. Battery modules can be removed for transport – the platform without battery modules weighs in at a manageable 100 pounds.

Structural “t-slot” aluminum is used for carriage construction. A reinforced box configuration uses aluminum plate to provide rigidity and protection from the elements. Standard ¼-20 socket-head fasteners are the primary means of assembly; with regular maintenance components secured using thumb knobs. Plates are mounted with flush fasteners to maximize clearance (ground clearance is designed to be at least 4.5” when traveling over soft ground). Casters are attached to the carriage using a robust rocker-arm system incorporating coil-over shock absorbers. These casters keep the platform upright during acceleration and while negotiating inclines. Removing two pins allows the caster assembly to swing up and out of the way for ease of access to battery modules, motors, and other chassis components.

Critical electronics are housed inside the chassis structure, protecting them from the elements. A tubular super-structure provides a flexible framework for mounting sensors, controls, and splashguards (required in wet conditions, as the cambered wheels cast off water toward the chassis). During challenge runs, the payload is secured to the top of the chassis structure with tabs designed for the purpose.

4.2 Electrical Subsystems Design

The electrical design is comprised of two largely independent subsystems. The 24V subsystem powers the drive motors and motor controller. The 12V subsystem powers the processors, sensors, and controls. Each battery module has two connectors, one for each battery. These connectors are mated to both the charging system and the chassis. For the 24V subsystem, circuitry on the chassis connects the batteries in series. For the 12V subsystem, circuitry on the chassis connects the batteries in parallel. The two subsystems share a common electrical ground reference.

4.2.1 Motors & Motor Controller

The motors are a pair of Midwest Motion Products model D33-455H-24V planetary gear motors, with a rated peak torque of 1062 in-lbs, and a continuous torque of 90 in-lbs at 5.1 amps. A 100 pulse-per-revolution encoder is affixed to the armature shaft. When combined with the 19.2:1 planetary gearing, this provides a detection capability of approximately 2mm over ground.

The motor controller is a RoboteQ model AX3500, capable of providing 60 amps of current to each of the motors. The controller is configured for analog input control on reset. Since there is nothing attached to the analog control inputs, a voltage divider internal to the controller keeps the voltage centered, which in turn causes the motors to be actively held steady. This capability is used to satisfy the E-stop requirements, as simply resetting the motor controller will cause the platform to come to a safe, controlled stop. A specific sequence of commands over the serial (RS-232) interface is required to switch the controller into serial operation mode. The controller has been pre-configured to use a closed-loop control while in the serial mode of operation, with a built-in top speed, configured for 10mph in this competition.

4.2.2 E-Stop

The wireless receiver has a range of over 150 ft, and can be paired with multiple wireless transmitter fobs. It incorporates a relay - the “normally closed” circuit of the relay is opened momentarily when the receiver is activated via wireless transmitter.

The E-stop design is shown in Figure 2. During normal operation, both the Manual E-stop switch and the wireless transmitter circuits are closed, causing the (active-low) reset input on the motor controller to be held high. If either the manual E-stop switch or the wireless receiver relay is opened, the reset input on the motor controller is pulled low by the pull-down resistor, and the motor controller is reset (into the analog input control state). The circuit includes sense connectors that allow the state of the E-stop system to be monitored by the processors. A circuit board was designed and constructed to implement this circuit.

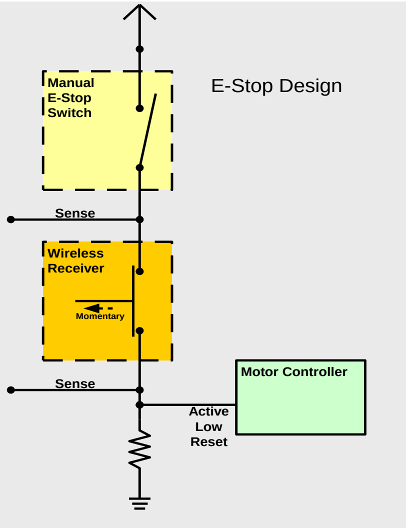


Figure 2 – E-Stop Design

While the manual E-stop switch is open, the motor controller is held in the reset state, and the motors are de-energized. The platform is easily maneuvered by hand while in this state.

Note: The wireless receiver has a second channel that is used to produce a one-button ‘start’ signal when initiating autonomous operation.

4.2.3 Power Conversion & Conditioning

The 12V subsystem incorporates three DC-DC 160W converters, designed for use as motherboard power supplies. Two of these are used for powering the computer systems; the third provides conditioned –12V, +12V, and +5V power to sensors and controls that require regulated power.

4.2.4 Processing Resources

The platform is configured with two computer systems. Each system is comprised of a microATX format motherboard, Intel Core i5 650 processor, 4GB of RAM, and 64GB solid-state drive. This particular microprocessor incorporates on-board graphics processing. When paired with a compatible motherboard the resulting system requires no additional graphics hardware, with the result that less power and space are required for the complete system. Each processor is housed in a Cooler Master slim case measuring 12.5” x 10.3” x 2.7”, a substantial reduction in space claim over previous years’ entries. The height of the case precludes installing PCIe cards directly into the motherboard, so the integrated power supply was removed from the case (since power is supplied via DC/DC conversion), and a PCIe riser installed in its place to accept the 4-port serial interface card used to communicate with RS-232-based peripherals.

4.2.5 Touch Screen

The Platform incorporates an 8” diagonal touch screen with 800x600 screen resolution. Power is supplied via the 12V (non-conditioned) power source, and communications is via USB 2.0.

4.2.6 General Purpose I/O

The Platform incorporates an ACCES I/O model USBP-II8IDO4A general-purpose I/O board for general purpose sensing and control. This board has 8 isolated digital inputs, 2 16-bit A/D inputs, and 4 solid-state (high-side FET) relays. The A/D inputs are used to monitor the 12V and 24V battery voltage levels for the purposes of estimating remaining available power. The solid-state relays are used to switch power for the safety light and horn. Digital inputs are used to sense E-stop circuit status as well as the reception of the start signal. Communications with the I/O board is over USB 2.0. Power is supplied via the 12V conditioned power source.

4.2.7 Safety Light

New for this year is the requirement that a safety light be illuminated while power is applied, as well as flashing during autonomous operation. We modified an inexpensive ‘emergency beacon’ to incorporate a standard socket for an automotive turn-signal lamp. The turn signal lamp has two circuits: one controls a low-intensity ‘parking lamp’ indicator, and the other a high-intensity ‘turn lamp’ indicator. The ‘parking’ indicator circuit is hard-wired into the platform power circuit so that it is illuminated at all times that power is available on the platform. The ‘turn’ indicator flashes under software control during autonomous operation (accomplished via a solid-state relay on the general-purpose I/O board). The lamp selected is LED-based and exhibits high-reliability, superior brightness, 360-degree visibility, and low power consumption attributes.

4.2.8 GPS

Platform position is obtained via a NovaTel ProPak-LB GPS receiver that incorporates differential corrections obtained via the OmniStar service for sub-meter accuracy in positioning. Communications with the GPS receiver is via RS-232. Power is from the 12V conditioned power source.

4.2.9 Electronic Compass

Platform heading is obtained via a PNI model TCM 3 electronic compass, selected for its configurability, sensitivity, and update rate. Communications with the electronic compass is via RS-232. Power is from the 5V conditioned power source.

4.2.10 Stereo Cameras

The platform incorporates two IEEE 1394 stereo cameras. Cameras supported include the Videre Design model STH-DCSG-6cm, and the Point Grey Research model BumbleBee2. Communications with and power to the stereo cameras are provided via FireWire.

4.2.11 WiFi Network Adapter

When required for the JAUS portion of the competition, the platform uses a Belkin Wireless G USB Network Adapter. Power and communications are via USB 2.0.

4.3 Platform Software

Platform processing components use Windows XP Professional as the operating system. Other than device drivers and device libraries provided by device vendors, platform software was developed using the Java language. The Java language was selected for its combination of portability, ease of development, and excellent performance characteristics.

Individual executable programs were developed for the autonomous, navigation, and JAUS challenges. A number of modules are common between two or more of the challenges. Figure 3 shows the general common architecture for the autonomous and navigation challenges.

Within the software group, a number of sub-groups were created, corresponding to major areas of processing. Interfaces were established between groups where appropriate, and development proceeded largely in parallel using agile software development techniques. Configuration management was via a secure Subversion server. A Java development environment with integrated Subversion client was used, making coordination of code between individuals and sub-groups a nearly trivial exercise. The sub-groups are as follows:

- Common Classes
- Device Interfaces
- Vision Processing & Obstacle Detection
- JAUS
- Global Path Planning – Autonomous
- Global Path Planning – Navigation
- Local Path Planning
- Simulation

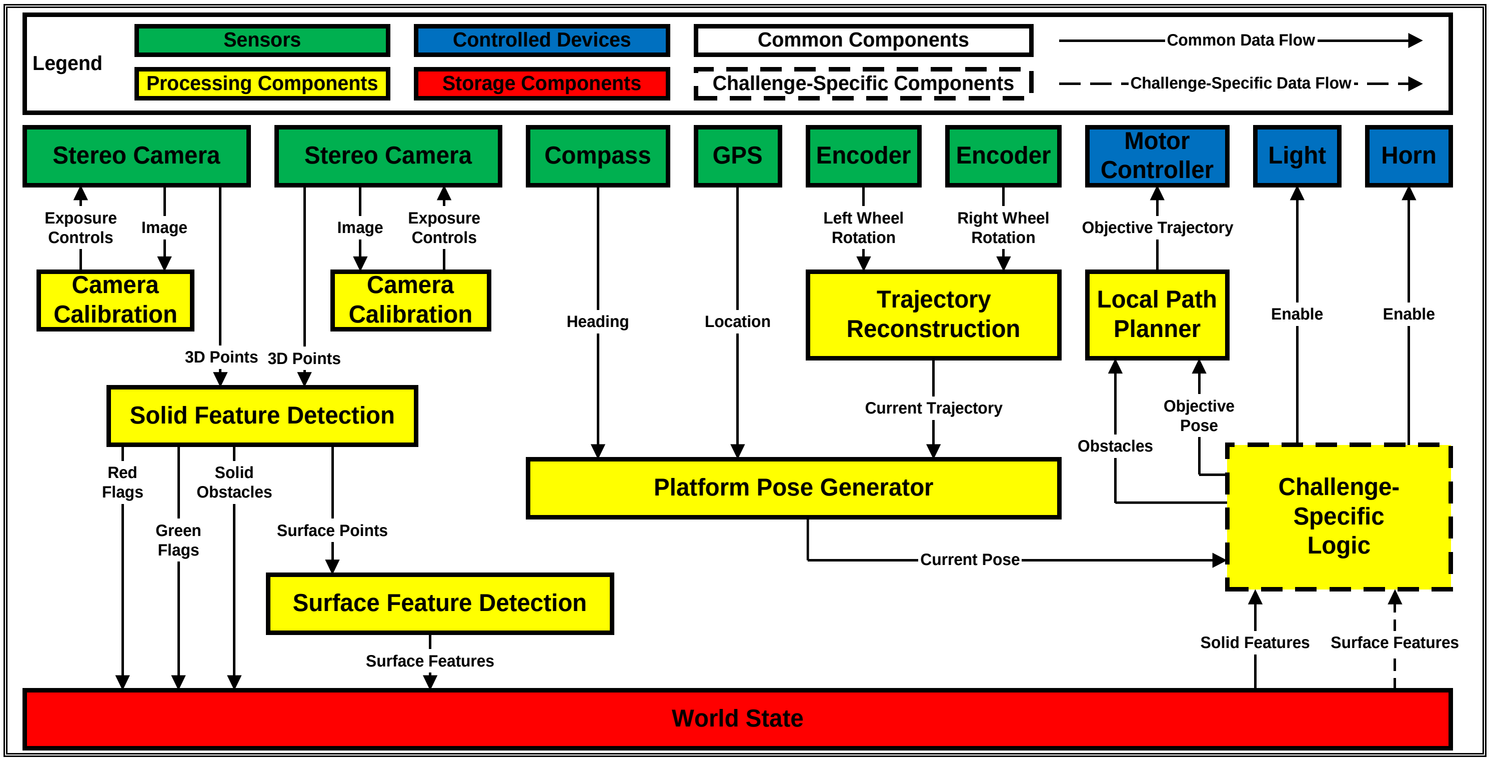


Figure 3 – Challenge Common Software Design

4.3.1 Common Classes

A number of common classes were developed for use by all developers where appropriate. This includes common-use patterns, data structures, dimensional, and geometric classes.

Two coordinate systems are used during processing. The 'local' coordinate system moves with the platform. Its origin is where the platform's natural axis of rotation intersects the ground plane. The positive X axis points forward, the positive Y axis points to the right, and the positive Z axis points into the ground. The 'global' coordinate system is fixed, with its origin located at some convenient point (a fixed latitude and longitude). In the global coordinate system, the positive X axis points north, the positive Y axis points east, and the positive Z axis points into the ground. Classes were developed that allow straightforward mapping between latitude/longitude and the global coordinate system, as well as between global and local coordinate systems.

4.3.2 Device Interfaces

Device-specific classes were developed. These classes interact with the devices over various communications interfaces, and convert between device-specific values and engineering units. At a minimum, the functionality required of each device was exposed as an interface.

4.3.3 Vision Processing & Obstacle Detection

This process analyzes the images from the stereo cameras to register objects and ground features into a local occupancy grid for the path planning analysis. The process consists of three steps: flag detection, solid obstacle detection, and surface feature detection.

Flag Detection

Each pixel of an image captured from the stereo camera includes RGB color information and (if stereo correlation for the point was successful) a XYZ coordinate representing the location in local 3-space from which the pixel color was sampled. The hue of each pixel in the image is evaluated against the range of acceptable colors for the red and green flags. The 3D positions of matching pixels are used to populate 3D occupancy grids. When all pixels have been processed, the occupied regions of the occupancy grids are analyzed to determine whether they match flag attributes (maximum diameter of six inches and between 18 and 30 inches above the ground plane). The locations of detected flags are converted to global coordinates stored in the global map as red or green flags. Figure 4 depicts the steps involved in flag detection.

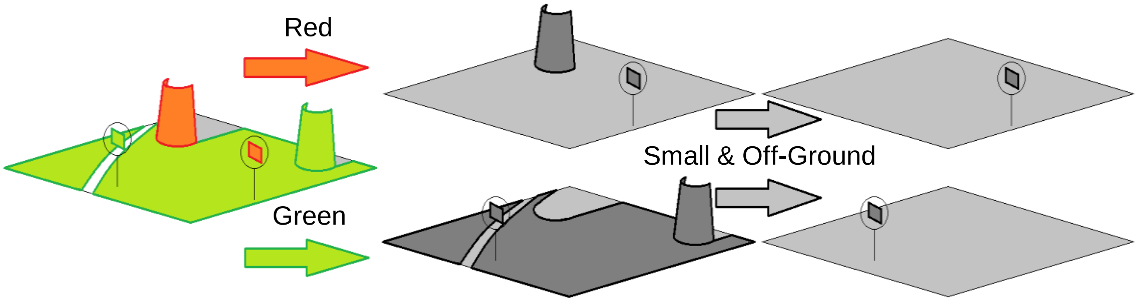


Figure 4 - Flag Detection

Solid Obstacle Detection

Each pixel in the stereo image is evaluated. Each pixel having a Z-coordinate that corresponds to a point above the ground plane is considered to be part of an obstacle and is used to populate a XY occupancy grid. When all points in the image are processed, outliers are scrubbed from the occupancy grid. The locations of grid elements that remain marked as occupied are converted to global coordinates and stored in the global map as solid obstacles. Each pixel in the image is then reevaluated and any pixel with an XY location in close proximity to an identified obstacle is marked as ineligible for consideration as a surface feature. This prevents the base of an obstacle from being interpreted as a surface feature. Note that flags are considered to be solid obstacles in addition to being flags. Figure 5 depicts the processing steps involved in solid obstacle detection.

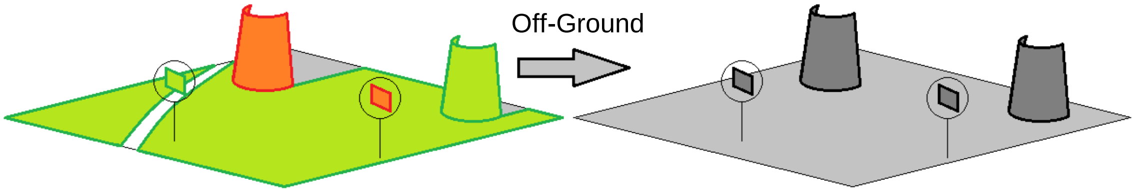


Figure 5 - Solid Obstacle Detection

Surface Feature Detection

In this phase, white markings on the ground are detected. All such marks of adequate size are considered obstacles. Those marks that exhibit the characteristics of lane boundaries are recognized and catalogued. To accomplish this, the pixels in the image are passed through a series of filters and transformations as follows:

The Roberts Edge Filter is applied on the image to extract the white lines from the image. Since the Roberts Edge Filter is noise sensitive, a blur effect is applied on the image before the edge detection. After applying the edge filter, the borders of the white lines adjacent to green grass manifest as a blue color in the resulting image. A subsequent filter converts the image to a gray-scale image, by selectively filtering for blue-hued pixels. Upon the completion of these processes, the image contains the white lines describing the boundaries of white surface features. These are stored into the global map as surface features. Additionally, those features having the characteristics of lanes are added to the global map along with lane orientation information. Figure 6 depicts the steps involved in surface feature detection.

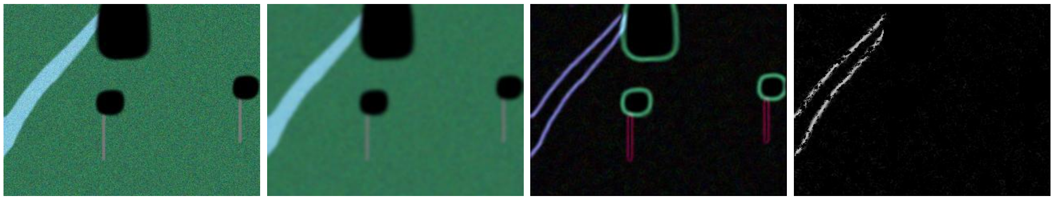


Figure 6 - Surface Feature Detection

4.3.4 Global Path Planning: Navigation

For the navigation challenge, a “Lifelong Planning A*” (LPA*) path planning algorithm was implemented. An A* path planning algorithm was considered, but ruled out in favor of the more efficient LPA* algorithm.

LPA* is an incremental heuristic search algorithm that combines both incremental and heuristic search techniques to speed up searches of sequences of similar search problems. This is well suited to the IGVC competition, in which target waypoints remain fixed in location. The algorithm recalculates only those possible paths that are invalidated when new obstacles are introduced that interfere with its current objective (composite) path.

The path-planning algorithm uses a ‘global grid’ data structure. This global grid is initially populated with the target waypoints, and all locations in the grid are considered to be traversable. An initial path is planned to navigate to all waypoints, minimizing distance traveled. As obstacles are discovered, this grid is updated with non-traversable locations. When these non-traversable locations intersect the planned path, re-planning occurs.

Initially, the set of ‘remaining’ waypoints is populated with the set of all waypoints. Incremental paths between each pair of remaining waypoints are generated along with associated cost (distance). All possible orders for visiting the remaining waypoints are generated. The costs of these ‘composite paths’ are calculated by summing the costs of the constituent incremental paths. These composite paths are placed in a data structure, ordered by increasing cost. The least cost path (at the head of the list) is always considered the objective path.

As the robot moves around the course, it will inevitably encounter obstacles that were unknown at the time of route calculation. New obstacles appearing on the global grid are checked for interference with the incremental paths. Where interference is detected, that incremental path will be reevaluated using LPA* and the new cost calculated. Composite paths that include that incremental path have their cost reevaluated, and their corresponding position in the ordered list updated as appropriate. When all affected composite paths have been reevaluated, the least cost path (at the head of the list) is once again used as the objective path. Table 1 shows Pseudo Code for the LPA* algorithm.

```
CalculateWaypointPaths();
PathPlan() {
    foreach ( OpenWaypointPath ){
        if( NewObstacleLocations == WaypointPath){
            RecalculateWaypointPath()
        }
    }
    CalculatePermutations();
    GetShortestPathFromPriorityQueue();
    SetLocalTarget();
    if( CurrentLocation == WayPoint ){
        CloseWaypoint();
        if( Waypoint == LastWaypoint ){
            exit();
        }
    }
    PathPlan()
}
```

Table 1 – LPA* Algorithm

When the platform position indicates it is within one meter of the objective waypoint, that waypoint is removed from the set of remaining waypoints. The robot continues to the next waypoint on its objective path unless a new obstacle is detected and a lower cost path becomes available.

Actual robot maneuvering is done via local path planning. The local path planner is provided with a vector value based on current robot position and the current optimal path plan. In instances where a section of the planned path is very linear, the magnitude of the passed vector is increased to reduce the number of identical vectors that have to be calculated. In areas where the planned path changes direction to avoid obstacles, the vector magnitude is greatly reduced to produce an array of vectors. This prevents the robot from making abrupt direction changes at obstacles and facilitates a smooth navigation path.

4.3.5 Global Path Planning: Autonomous

With the addition of waypoints to this year’s autonomous challenge, it becomes very like the navigation challenge in many respects. As such, the same general algorithm is used as for the navigation challenge, with the following exceptions:

1. The order of waypoints is fixed, so the composite path is always known.
2. White markings on the ground are interpreted as obstacles.
3. Detected flags cause ‘synthetic’ obstacles to be generated extending from the flag locations outward to the corresponding lane boundary.

4.3.6 Local Path Planning

After a global path has been planned with a global path-planning algorithm a smaller section of the global grid centered on the robot is processed with a local path-planning algorithm that determines the objective trajectory (velocity and angular velocity) of the platform.

The local path-planning algorithm uses vector field techniques to arrive at this objective trajectory. Occupied regions of the grid contribute vectors oriented toward the platform position with a magnitude that is inversely proportional to the distance of the occupied region from the platform. The current objective position (as provided by the global path planning component) contributes a vector oriented toward the objective position. (It is the job of the global path planning algorithm to detect dead-ends and to adjust the objective position so as to avoid them.) The average of these vectors is used to determine an objective heading and velocity. An objective trajectory is calculated that quickly aligns the platform heading and velocity with the objective heading and velocity.

As the platform is constantly moving, the objective heading and velocity are constantly changing, and so too is the objective trajectory changing. The result is a dynamic, self-stabilizing approach to local path-planning.

4.3.7 Simulation

During the early stages of the project, we needed a way to test our path planning and driving algorithms before we had a physical robot ready. We selected an open-source software package called Player/Stage to enable us to simulate our robot's behavior in an obstacle-rich environment.

Player/Stage

Player/Stage is a real-time simulator for robots that models their drive motors, range-finding sensors, and GPS. We chose Player/Stage because it encompasses all of the elements we needed to simulate, we could interface with it using the same programming language in which we wrote our navigation program, and because its widespread adoption by well-known robotics firms and government research agencies led us to believe that it would be stable and practical.

Simulation Process

We interfaced Player/Stage with our preliminary driving program using an adapted version of the program that drives our robot and processes input from the sensors. Instead of sending commands to motors and getting data from our image processing code, it communicates with Player/Stage to command and monitor the virtual robot. From our program's perspective, there is no distinction between controlling the simulator as opposed to an actual robot.

To test our navigation algorithm, we recreated the navigation challenge's waypoints and obstacles on the simulator's map and ran our virtual robot through the course. This simple, swift testing process allowed us to tweak, fix, and optimize our program, minimizing the amount of work we needed to do when we completed the mechanical robot.

Challenges

While Player/Stage lent us a distinct advantage during the development process, it also presented us with a number of challenges. Despite its popularity, there is little documentation on performing even simple tasks. As a result, programming the simulator interface took significantly longer than it otherwise would have.

Additionally, we could not simulate our robot's characteristics with complete accuracy. Due to the two-dimensional nature of the program, we had to simulate vision with a laser rangefinder instead of a camera. The rangefinder gave us accurate readings and provided us with a good way to simulate our pathfinding algorithm, but we had no means of analyzing the effects of any inaccuracies or artifacts that our camera vision system might produce. We also had to use approximations for a number of physical constants (such as the robot's maximum acceleration) that have a significant impact on the behavior of our driving algorithm.

4.3.8 JAUS

For this year's competition the JAUS code had been retooled from the previous year's entry. This year's improvements include proper handling of reserved values. Additionally, the JAUS software control structure has been inverted, allowing the JAUS code to be used as a module rather than the main program. For testing, a rudimentary Common Operating Picture (COP) application was developed that allows testing of individual message sets. For debugging purposes, the Wireshark application was used in conjunction with the COP to ascertain the flow of packets and verify the format of messages coming from the JAUS module.

UDP packets are used to send JAUS messages between the platform and the COP. Once the JAUS module receives a message it parses the message into header and the payload, which are each then checked for validity. The details of valid messages are passed to the proper JAUS module to handle the request.

5. Predicted Performance

The following points describe predicted performance and the methods used to determine these numbers.

- **Speed –** Propulsion system design and component selection were undertaken with a goal of achieving the maximum allowed speed of 10 miles per hour under anticipated conditions. This top speed is enforced by setting the motor controller into a PID feedback loop configuration, and by adjusting the encoder scaling parameters appropriately.
- **Ramp climbing –** Propulsion system design and component selection were undertaken with a goal of performing at top speed on a 15% gradient, the specified maximum under IGVC rules. Performance to this goal has been verified in trials.
- **Reaction times –** The vision system achieves a sustained throughput of 15 frames per second for each camera. Based on an analysis of latency in acquisition, processing, and communications paths, it is estimated that an obstacle presented within the field of effect will affect motor speed in 75 +/- 25 milliseconds.
- **Battery Life –** Battery life is highly dependent upon the operational environment. Under continuous load and with a full charge, the 24V battery module life is estimated at 2 hours. The 12V battery life is estimated at 5 hours under full processing and sensor load.
- **Obstacle Detection Distance –** This is configurable via parameters to the stereo vision processing software. Detection is presently limited to 6 meters.
- **Complex Obstacle Negotiation –** Switchbacks and traps are handled as a natural consequence of the path planning algorithm, described in the software design section.
- **Navigational Accuracy –** The geolocation equipment used is capable of sub-meter accuracy when used with satellite- or earth-based augmentation. The GPS sensor is capable of employing satellite-based augmentation and is presently configured to use the OmniSTAR service for differential corrections, which after initial settling will generally achieves a standard deviation of 0.3 meters or less from actual.

6. Cost Data (in dollars)

Mechanical / Propulsion		Sensors		Processing / Electrical	
Wheels, hubs, and tires	500	Stereo Cameras	1,755	Computer Systems	1500
Chain & Sprockets	150	GPS System	2,700	Power Supplies	150
Aluminum/Steel Stock	950	Compass	775	E-Stop System	40
Miscellaneous Hardware	250	Motor Controller	410	Safety Light	40
Batteries	340			Touch Screen	315
Motors	1,790			Misc. Electrical	200
Shocks & Casters	160				
				TOTAL	$11,985

7. Labor Data (in man hours)

Mechanical		Electrical		Software	
Design	180	Design	40	Device Interfaces	90
Fabrication	240	Component Selection	20	JAUS-Specific	120
		Integration	40	Algorithm Development	420
				Administrative	180
				TOTAL	1330