CAPSTONE DESIGN
Fall 2007

INTELLIGENT GROUND VEHICLE COMPETITION
TEAM 2007-2008

Abstract: The Intelligent Ground Vehicle Competition challenges university students around the world to design and manufacture an autonomous robot. The vehicles are to participate in a series of events competing towards the grand prize. The following report contains a detailed design breakdown of the 2008 LSU Intelligent Ground Vehicle. The report justifies all design decisions made throughout a four-month design process in both the mechanical and electrical engineering teams. It also provides manufacturing, testing plans and safety preventions for the robot.

ME Group Members:
David Mustain
Diego Gonzalez

ECE Group Members:
Alex Ray
William Burke
Geoff Donaldson

Advisors:
Bryan Audiffred
Dr. M. S. de Queiroz

Date of Submission: December 7, 2007

Table of Contents

(i) EXECUTIVE SUMMARY

The Intelligent Ground Vehicle Competition (IGVC) is an annual event focused on the design and development of automated vehicle technology. The competition is divided into different events where each team has to face a different challenge. All vehicle operations (including obstacle detection, mobility, path following and GPS Navigation) are required to be entirely autonomous. The 2008 competition will be held in Rochester, Michigan on May 30.

Louisiana State University is planning to participate in the upcoming IGVC with a team of CAPSTONE design seniors. The team is comprised of two Mechanical Engineers (David Mustain and Diego Gonzalez) and three Electrical and Computer Engineers (Alex Ray, Geoff Donaldson and William Burke). This design team is determined to overcome the difficulties that have prevented previous LSU teams from competing.

The LSU IGVC team has followed a four-month design process and devised a successful prototype that will be ready for fabrication within the next month. The first step in the design process was to establish the following goals and objectives:

- ⇒ Dampen vibrations induced on the camera pole of the previous IGVC vehicle
- ⇒ Design a new frame with better weight distribution and functionality than previous IGVC vehicle
- ⇒ Devise effective algorithms for automated navigation
- ⇒ Ensure that the vehicle can perform all of its functions when needed
- ⇒ Design and build a vehicle that meets all the qualification requirements and is able to compete
- ⇒ Successfully complete at least two out of the three events

Next, The design of the vehicle was separated into eight categories. A brief overview of each is discussed below. The enclosed report includes a detailed description of each of these areas as well as information on testing, manufacturing, safety and costs.

1. Power

Electrical power is an absolute necessity for this project. Since the robot can have no connection to external power, six batteries will be carried onboard. These batteries will supply power to the motors and all of the computer equipment.

2. Obstacle and Boundary Detection

The robot will utilize a unique custom obstacle and boundary detection algorithm that will combine input from a digital camera mounted approximately six feet above the ground, a laser range founder that scans a horizontal plane about a foot off the ground, and emergency distance sensors positioned around the robot's perimeter. The computer will gather data from all of these sensors and create an artificial overhead view of the course ahead in a weighted grid form so that the path planning algorithm can find the most appropriate path.

3. Path Planning

The path planning algorithm consists of two modes. The first mode, long distance path planning, will receive the overhead grid from the obstacle and boundary detection algorithm and map out the best path through the obstacles according to the robot's knowledge of the course at the time. The robot will then transition into the second mode, short distance path planning, which will execute the path generated by the first mode while verifying that the way is clear of obstacles, boundary lines, and potholes. The system will periodically re-execute the long distance path planning mode to maintain a continually fresh view of the course.

4. Computer Processing

The computer system must interface to the sensors and motor control and perform all the necessary calculations for the successful operation of the robot. This includes interfacing communication protocols for each sensor, using the sensor's native software, or developing new software to integrate each sensor into custom algorithms. A small form factor computer was chosen to accomplish this task, along with a dual core CPU and a powerful graphics card to implement GPU offloading. GPU offloading involves porting the computer vision algorithms to the graphics card, which is more suited to performing the independent floating point operations that are often numerous in computer vision processing.

5. Global Positioning System (GPS)

In addition to the camera, laser range finder, and emergency distance sensors, the robot will employ a differential GPS unit. This is necessary for the navigation challenge, but will also be used in the execution of the proposed path during the autonomous challenge.

6. Motor Control

The robot's motor control system will be implemented using an off-the-shelf printed circuit board solution. The software will keep a command queue in memory and pass each command to the motor control board in sequence. The turning will be accomplished through differential steering. The motor control also features an emergency stop input and a hardware speed limiter, which complies with the rules of the competition.

7. Chassis Design

The chassis for the robot will be made out of aluminum 6061 due to its low-weight and high strength. It will consist of three levels, which will hold different components. The first level will hold the heavy equipment in order to lower the center of gravity of the vehicle. The second level will hold a drawer in both the front and the back to allow for easy access to electronic

components. Finally, the third level will provide weatherproofing for the electronics with the use of metal sheets.

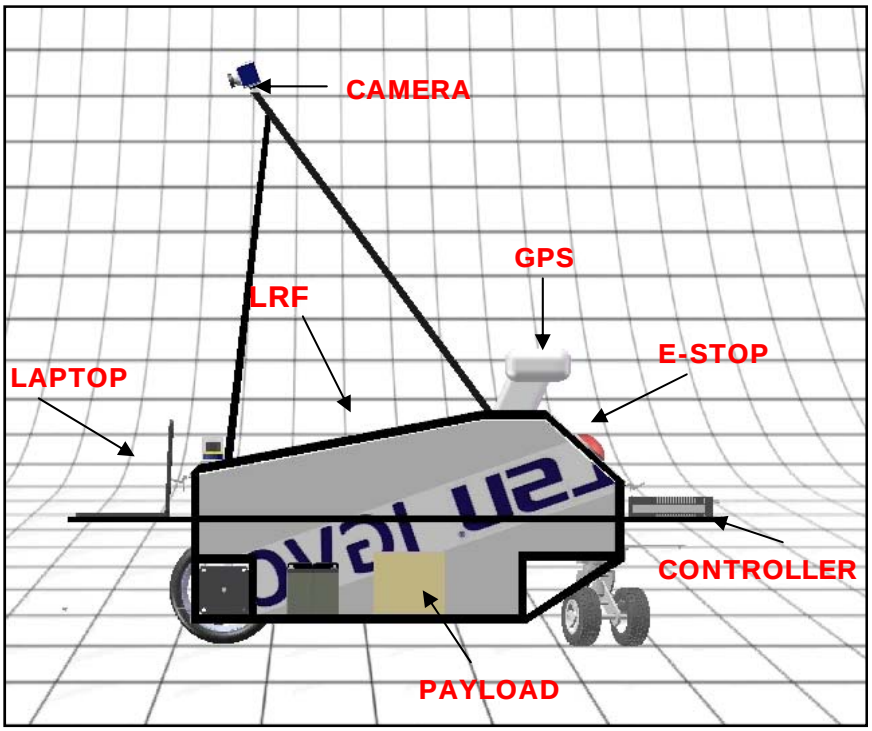


Figure 1: *Component configuration on vehicle*

8. Vibrations

Vibrations transmitted to the camera will result in poor image detection and make computer vision more difficult. The camera will be rigidly attached to the frame of the robot via a bolt on tripod. Vibrations transmitted to the frame, and hence the camera, will be reduced by using low air pressure in the front tires and by the addition of a vibration damping caster design in the rear. The caster will use two pneumatic tires attached to a standard spring caster bearing. The springs will be designed to match the spring rate of the front tires, which will be experimentally determined in the Spring semester.

(1) BACKGROUND INFORMATION

The Association for Unmanned Vehicles International (AUVSI) is the world's largest non-profit organization and the sponsor of several autonomous vehicle competitions each year [1]. These competitions challenge college students to design and build autonomous ground, air and marine vehicles. The vehicles are required to be entirely autonomous, meaning that no human intervention in its operations is allowed. Furthermore, the vehicles are tested in several events for obstacle avoidance, design and navigation.

The Intelligent Ground Vehicle Competition (IGVC) hosted by AUVSI offers the opportunity for Mechanical, Electrical and Computer Science Engineers to work together in the design of an intelligent robot [2]. This yearly competition requires teams to design and manufacture a vehicle to complete an obstacle course where the vehicles are awarded points for obstacle avoidance and lane following. The second event in which the teams are required to compete is based on GPS waypoint navigation. Finally, the teams must provide a design report and presentation that will also award them points towards the grand prize. The next IGVC will be held in Rochester, Michigan on May 30, 2008. An overview of the competition rules and events is provided in this section of the report (detailed information can be found at http://www.igvc.org [2]).

(1.1) Competition Rules [3]

In order to promote fair competition, IGVC has devised a set of competition rules. All teams must conform to the following specifications:

- ⇒ **Design**: Must be a ground vehicle (propelled by direct mechanical contact to the ground such as wheels, tracks, pods).
- ⇒ **Length**: Minimum length three feet, maximum length seven feet.

⇒ **Width**: Minimum width two feet, maximum width five feet.

⇒ **Height**: Not to exceed 6 feet (excluding emergency stop antenna).

⇒ **Propulsion**: Vehicle power must be generated onboard.

⇒ **Speed:** Maximum vehicle speed of five miles per hour (5 mph) will be enforced. All vehicles must be hardware governed not to exceed this maximum speed.

⇒ **Mechanical E-stop location**: The E-stop button must be a push to stop, red in color and a minimum of one inch in diameter.

⇒ **Wireless E-Stop:** The wireless E-Stop must be effective for a minimum of 50 feet.

⇒ **Payload**: Each vehicle will be required to carry a 20-pound payload. The shape and size is approximately that of an 18" x 8" x 8" cinder block.

(1.2) Autonomous Challenge

This event of the competition requires the intelligent vehicles to complete an obstacle course under a prescribed time while staying within the 5 mph speed limit and avoiding the obstacles on the track. Judges will award points based on shortest time taken and ability to avoid obstacles.

The course will be laid out on grass, pavement, simulated pavement, or any combination, and be 700 to 800 feet in length. Continuous or dashed white and/or yellow lane lines painted on the ground will designate the course boundaries. The track width will be approximately ten feet wide with a turning radius not less than five feet.

The course will contain obstacles such as artificial inclines with gradients not to exceed 15°, sand pits (sand depth 2 - 3 inches), potholes and randomly placed 5-gallon pails, construction drums, cones, pedestals and barricades that are used on roadways and highways.

(1.3) Navigation Challenge

This event requires for the intelligent vehicle to autonomously travel from a starting point to a number of target destinations (waypoints or landmarks) and return to home base, given only the coordinates of the targets in latitude and longitude. Coordinates for the actual navigation course waypoints will be given to the contestants on June 7, 2008 in degrees latitude and longitude. Construction barrels, barricades, fences, and certain other obstacles will be located on the course in such positions that they must be circumvented to reach the waypoints. These may be randomly moved between runs. Also, the course will be divided into two areas by a fence with a 2 meter wide opening located somewhere along it (no coordinates are provided). The opening will be randomly relocated along the fence at the start of each run.

(1.4) Literature Review/ Competitor Vehicle Analysis [2]

The following section provides a summary of the vehicles used by the top four universities in last year's competition. These vehicles were chosen due to their proven success during the past. It is important to note that all teams used a digital camera, laser range finder, differential GPS, and a digital compass, which are all components that the LSU 2008 IGV will be using.

#1 Johnny-5 Virginia Tech

- Equipment
  - Digital Camera
    - 640x480 resolution
    - 94 degrees field of view
    - 15 FPS
  - SICK LMS-221 Laser Range Finder
    - 180 degree field of view
    - 3.2ft – 65.6ft range
  - Differential GPS
  - Digital Compass
    - Resolution 0.1 degrees

The Johnny-5 team wrote custom device drivers to ensure that only the latest frame from the laser range finder and digital camera were sent to the algorithms. They translated all of the data to a common coordinate system and combined it to form a map of the course. The addition to last year's entry was a Kalman Filter, which the team used to predict and update the path from the GPS. This enabled them to use a much cheaper and less accurate GPS unit. Their path planning algorithm was to examine 36 possible parabolic paths beginning with the one closest to the current boundary line. The first path that did not hit an obstacle was taken.

#2 Awesome-O University of Minnesota

- Equipment
  - Digital Camera
    - 30 FPS
    - Fish-eye lens
    - 160 degree field of view
  - SICK LMS Laser Range Finder
    - 5m range
  - Differential GPS
  - Magnetic compass

The University of Minnesota team used an 81x81 sized grid for their path planning algorithm. Obstacles in the grid were represented by a 1. Once the obstacles were placed, the grid was smoothed with a Gaussian filter to add a cushion area. Then they used a shortest path algorithm to find the lowest weight path between two points on the grid.

They took the approach of offloading their graphics processing onto the GPU using the OpenVIDIA library. They first masked out the horizon and the robot from the image, thresholded the points, and ran a particle filter over the image to eliminate extraneous small particles. At this point, they did fish-eye lens correction, followed by blob removal, and a Hough transform. Finally, they constructed the weighted matrix. The path planning algorithm used the A-star algorithm with an added heading reward to make the robot tend to move straight forward instead of left and right.

#3 Capacitops University of Detroit Mercy

- Equipment
  - Digital Camera
    - 144-79 degrees field of view
  - SICK LMS 200
  - Differential GPS
  - Digital Compass
    - 0.5 degrees resolution
    - 20 Hz update frequency

Capacitops used the Player/Stage Linux framework to control the robot from a server. The team ran both a server and client pc on the robot. They utilized the OpenCV vision library from Intel. For image processing, they weighted the grayscale conversion towards the blue channel due to its natural contrast between obstacles, lines, and grass. They used a simple Sobel edge detection scheme and morphological open operations to connect the lines. Finally, the Hough transform was used to parameterize the lines. Navigation used a Kalman filter, though the use was not fully explained. The robot also incorporated a side to side scan in the event that it lost its way.

#4 Polaris Virginia Tech

- Equipment
  - BumbleBee Stereo Vision
    - Retail stereo vision in-a-box system
  - Sick LMS 221
  - DGPS
  - Digital Compass

Polaris used a simple brightest pixel threshold operation to segment out lines for a Hough transform. They also translated all data to a common coordinate set. They also examined potential paths like Johnny-5.

Overall, the vision algorithms used by these teams are fairly mundane. Polaris' method is the simplest of the four surveyed, and it was also the lowest ranked. It is difficult to tell why

Polaris ranked lower and Johnny-5, because most of the algorithms they described in the required design reports were the same. Computer vision is an untapped resource for these teams, and a more advanced algorithm will give the LSU entry an edge. Because of the simplicity of the vision processing, the path planning of these teams is probably not as sophisticated as that which is planned for the LSU robot. The improvements over competing designs bodes well for MikeRobot's success in the competition.

(2) PROTOTYPE DESIGN

The design of the 2008 LSU Intelligent Ground Vehicle (MikeRobot) was divided between Mechanical (ME) and Electrical (ECE) teams. Obstacle detection, line detection, wave point navigation and the evaluation of best routes are some examples of tasks that were addressed by the ECE team. Although the electrical focus varied from the mechanical one, it was important that the design of the vehicle operated as one unit. For this reason, the mechanical design was molded around the needs of the electrical design. This included vibration reduction, maneuverability, power distribution, weight distribution, structural integrity and weatherproofing.

This section of the report states the engineering specifications and critical issues of the vehicle's design. The final concept is then presented with proper justification of how the requirements of the project were met.

(2.1) Constraints and Engineering Specifications

The main constraints of the vehicle's design were established by the rules of the competition (Refer to section 1.1). From these rules, the constraints that significantly affected the design were the size limitations and the speed control. From these general constrains, a

series of more specific constraints were developed and expanded using engineering specifications:

A. Physical Requirements

(i) Weight:

-The assembled vehicle should weigh less than 250 lbs. including all components.

(ii) Size:

-The vehicle should be as small as possible within the competition's requirements. A smaller size will provide easier navigation around obstacles.

(iii) Performance

-The vehicle must operate autonomously.

-The vehicle should be able to travel at a speed of 5 mph

-The vehicle should be able to drive up and down inclines with a 15° gradient while maintaining a constant speed.

(iv) Stability

-The vehicle should be able to turn at a minimum two feet radius at 5 mph

-Electronic equipment on-board should be protected in case of rollover

(v) Environmental

-The vehicle should be able to operate at temperatures ranging from 25 -110 degrees Fahrenheit.

-The vehicle should be able to operate in light rain and overcast conditions.

-The vehicle should be able to navigate on terrains such as: pavement, concrete, grass, gravel and dirt.

B. Power Requirements

(i) Batteries should be rechargeable

(ii) The motor power source should also be used to power other electronics.

(iii) The vehicle should have enough power for 30 minutes of autonomous navigation

C. Safety

(i) The vehicle should have a mechanical and wireless emergency stop

(ii) The wireless e-stop should be able to reach a distance of 50 feet

(iii) The motor controller should limit the speed to 5 mph

(2.2) Obstacle Detection

The tasks of obstacle detection and boundary detection are accomplished in tandem in the design. The method uses the combination of a camera, laser range finder, and knowledge of the course and obstacles to reliably create a model of the course to pass to the path planning algorithm. Example images are included, but they are only relatively similar to what will be seen by the camera. They were taken by a standing person at a horizontal angle. The IGV camera will be about the same height and angled down towards the ground, so any horizon will not be present in the actual image. The method will operate as follows:

1. Camera Distortion Correction

Most digital cameras are equipped with lenses that cause a significant amount of distortion to the image. When the image is used for computer vision purposes, the effect of this distortion is especially relevant. There exist models to account for the distortion, and the model in equation (1) was chosen for this design.

$$\begin{bmatrix} x_d \\ y_d \end{bmatrix} = (1 + k_1 r^2 + k_2 r^4 + k_5 r^6) \begin{bmatrix} x_n \\ y_n \end{bmatrix} + \begin{bmatrix} 2k_3 xy + k_4(r^2 + 2x^2) \\ k_3(r^2 + 2y^2) + 2k_4 xy \end{bmatrix} \qquad (1)$$

The model includes both radial and tangential distortion. The radial distortion accounts for the barrel shape of the image, and tangential distortion accounts for the centering of the radial distortion. The Matlab Camera Calibration Toolbox was used to fit constants to the distortion model from twenty-five images of a checkerboard pattern. The analysis revealed that the principle cause of the distortion was radial, and the tangential component was negligible. Therefore, the simplified model in equation (2) was appropriate.

$$\begin{bmatrix} x_d \\ y_d \end{bmatrix} = (1 + k_1 r^2 + k_2 r^4) \begin{bmatrix} x_n \\ y_n \end{bmatrix} \qquad (2)$$

Because the simplified model is still a high order polynomial, there exists no method for finding an inverse function. Therefore, numerical methods must be used to correct for the distortion. To avoid performing redundant calculations inside the robot's vision loop, the distortion correction will be calculated offline and stored in a lookup table. Results of image correction using the values obtained by the Matlab Calibration Toolbox are shown in **Figures 2.2.1** and **2.2.2**. The corrected image is noticeably cropped by the toolbox. The actual image will not be cropped in such a way so as to not lose any information from the image.

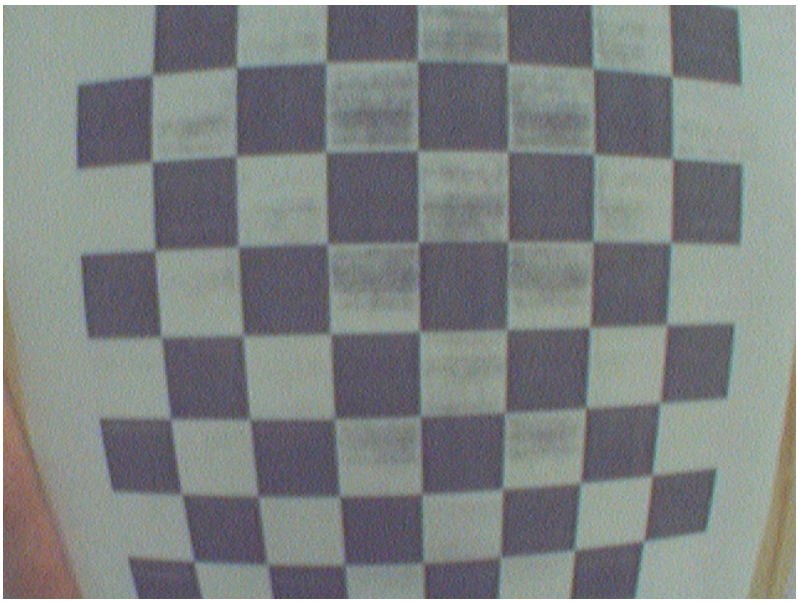


Figure 2.2.1: *Image from camera without distortion correction*

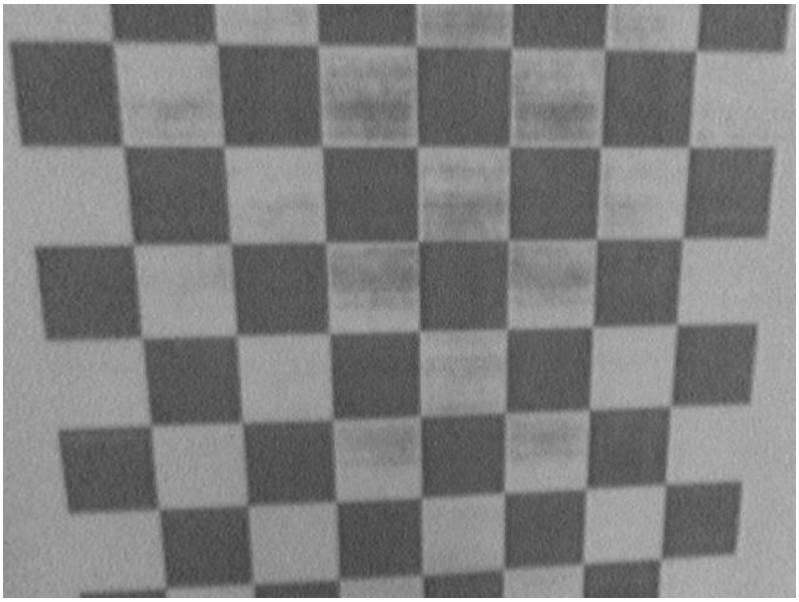


Figure 2.2.2: *Image from camera with distortion correction*

2. Image Color Identification

Once the barrel distortion has been corrected, the colors in the image are identified. Human beings name colors using various different distinguishing categories, such as green, blue, red, and yellow. The number of categories varies from language to language and culture to culture, but we will be using the eleven names commonly accepted in English: black, blue,

brown, grey, green, orange, pink, purple, red, white, and yellow. The color naming algorithm, which is described in *Learning Color Names from Real-World Images* [4], uses a trained matrix that was constructed from image sets from Google Image Search and Ebay to easily identify the color of a pixel in question. The algorithm converts the RGB values to L*a*b colorspace and finds the appropriate name for the color using the lookup table. The code used was supplied by the authors of the paper on their website [5]. The code can produce either a colorized image or an indexed image as an output. The indexed image is much faster to execute and more suitable for the intelligent vehicle's purposes. **Figures 2.2.3** and **2.2.4** show an example original image and color-identified image respectively.

Figure 2.2.3: *Representative Image from IGVC course [2]*

Figure 2.2.4: *Image after color identification*

3. Color Segmentation

The main reason to name the colors in the image is to facilitate partitioning of the image based on color. Some obstacles in the course are known to be white, orange, yellow, and blue, but more importantly, the majority of the course will be run on grass, which is obviously green. Preliminary tests with the color naming algorithm using pictures from the IGVC website of previous competitions show that the grass is easily categorized. Therefore, a mask can be made to remove all of the green-colored pixels from the image. This not only generates excellent segmentation between obstacles, but also reduces the information of the image so that the processing will be less complicated. The color segmentation step is extremely fast and requires only a few operations at most. The algorithm will simply test to see if the pixel is of any of the colors specified and delete it if the color is not included in the list. There are many directions that the algorithm can take at this point, but the most appropriate for the team's purposes is to generate a binary image that includes both ground features, such as boundary lines and potholes, and obstacles.

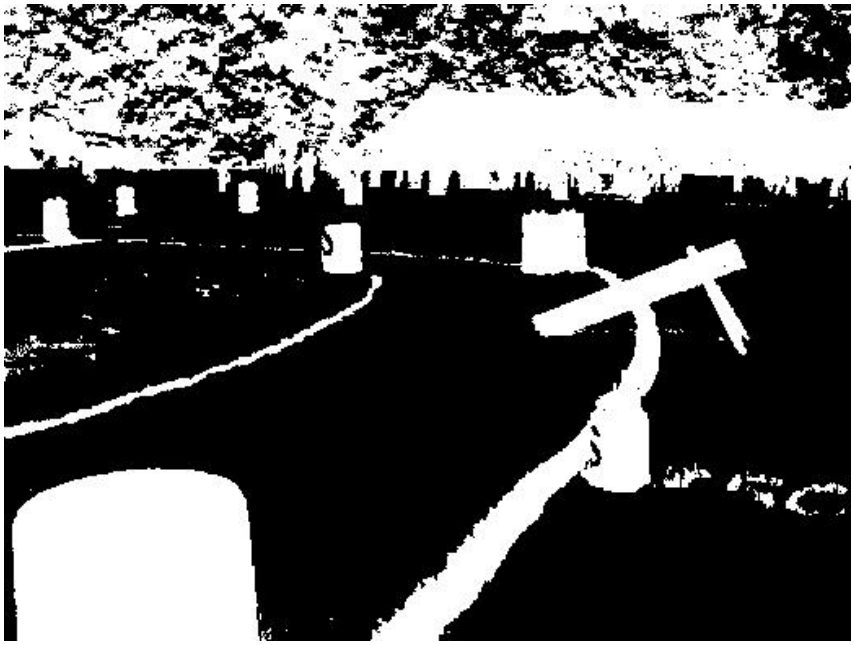


Figure 2.2.5: *Binary image showing pixels that are not green*

4. Image Obstacle Placement using Laser Range Finder Data

Once the obstacles and ground features have been segmented from the background of the image, the position of the obstacles in the X-Y plane should be determined. There are two coordinate systems in question at this point: world coordinates and image coordinates. The former refers to the global position of objects in three dimensions. The origin can be placed anywhere, and the most logical place is directly beneath the laser range finder. Image coordinates refers to a location in pixels on the image that corresponds to a point in world coordinates. Using standard camera geometry methods and the pinhole model for a digital camera, the mapping from world coordinates to image coordinates is trivial. The operation is shown in equation (3).

$$x = \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = C \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} R & T \\ 0 \dots & 1 \end{bmatrix} \begin{bmatrix} X_w \\ Y_w \\ Z_w \end{bmatrix} \qquad (3)$$

Figure 2.2.6 shows the relationship between world and camera coordinates.

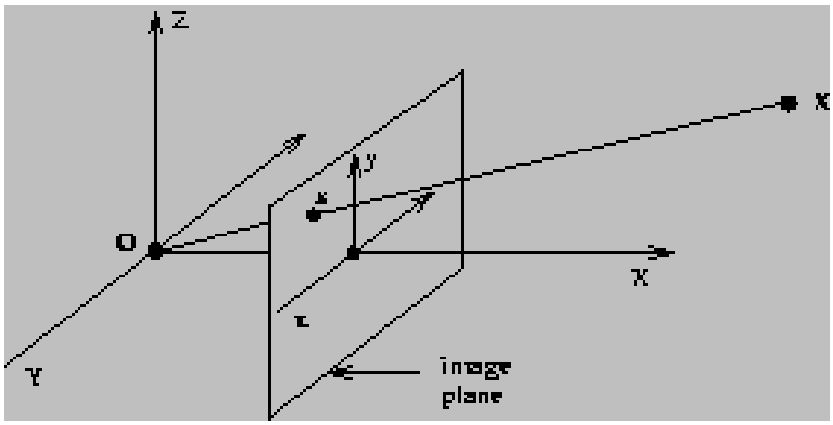


Figure 2.2.6: *Mapping of real world coordinates to image plane*

Unfortunately, the operation is not immediately reversible, because the 3x4 matrix has no inverse. To translate from image coordinates to exact world coordinates is impossible, because any point on a specific line in world coordinates can be projected to the same point in image coordinates. To achieve a decisive placement of the object in the X-Y plane requires more information.

The extra information comes from the laser range finder. Because the location of the laser range finder is fixed compared to the world coordinates origin, its X, Y, and Z world coordinates are known. If the laser range finder is taken to be at <0,0,Z>, then the values it reports will actually be exact three dimensional world coordinates. These world coordinates can be mapped into image coordinates precisely using equation (3).

Once the corresponding image coordinates are known, the point can be matched to the segmented blob of color that was previously identified as an obstacle or ground feature. If any of the world coordinates X, Y, or Z is known for a point in image coordinates, then equation (3) will reduce to an equation that contains a 3x3 invertible matrix. Therefore, transformation from image to world coordinates becomes possible.

Because the camera will be taller than all obstacles on the course, the image will contain both the obstacle's leading face and its top. This will actually yield a method to calculate the

depth of the obstacle as well as its position. To place the obstacle then requires two assumptions:

- Obstacles are planar on both front and rear faces with respect to Z. This means that if Y is held constant, all X values above and below the point reported by the laser range finder will be constant and also that the topmost pixel in the image represents the farthest extent in the X direction of the object.
- All obstacles have the same height H.

The first assumption only fails with any meaningful effect when it comes to A-frame barriers, so these are treated slightly differently. An example is shown in **Figure 2.2.7**. The algorithm will use a solidity analysis to differentiate between the A-frames and other obstacles. This entails finding holes in the object's frame which will be present in the binary image. There are Matlab functions to quantify the presence of these holes, and these can either be used directly or emulated. Once the object has been classified as an A-frame, a width measurement will be used to attempt to ascertain the A-frame's orientation.

Figure 2.2.7: *Example of A-Frame IGVC Obstacle [2]*

Other obstacles will be treated in the following manner. Since the height of the obstacles is assumed to be constant, and the front plane is assumed to be planar with respect to

Z, the location of the top of the obstacle can be calculated based on the X and Y values obtained from the laser range finder. This location will be a precise set of world coordinates, so it can be mapped to the image, and will project inside the color area of the obstacle. If the point does in fact represent the top of the leading face of the obstacle, then the remainder of the color area above the projected point will represent the visible top of the object. Therefore, the depth of the obstacle can be deduced by measuring the distance from the estimated top of the leading face to the edge of the color area.

5. Image Ground Plane Segmentation using Laser Range Finder Data

If the laser range finder is placed in such a way that no obstacle will have a height shorter than its scan height, then this information can be used to segment out the ground plane. Since the laser range finder will report the first object that it encounters, it can be assumed that all area with X values less than that object have Z coordinate equal to zero. Therefore once the original points reported by the laser range finder are mapped to the image, a position for the base of the object can be calculated based on the assumption of a leading edge planar obstacle. Once this point is transformed from world coordinates to image coordinates, it will be known to be the beginning of the rise of the object. Since the object is the first thing to obstruct the view of the laser range finder, it can be further deduced that all points along a straight line from the object's base to the world coordinate's transformed location in image coordinates have Z=0 in world coordinates. In other words, everything between the laser range finder and the first object that it detects is assumed to be the ground.

With only the image, it is impossible to tell whether the area pictured is actually at ground level or rises above the ground plane, but with the additional laser range finder data, the whole region can be mapped to the Z=0 plane in world coordinates. This has the very attractive

effect of allowing the algorithm to examine only the ground for boundary and pothole detection or filter out any extraneous boundary lines and ground features that might complicate obstacle placement. For example, a sandpit will probably cover the entire course at some point during the run. If there were an obstacle placed on top of the sand, the placement of that obstacle might be difficult. However, with the option of ground plane segmentation, the sand pit can be identified as a ground feature, allowing the boundary algorithm to focus on its identification.

6. Image Ground Plane Segmentation using Color Data

The only drawback to using the laser range finder data to identify the ground plane is that the beam is blocked by the first obstacle. Therefore, though there may be more obstacles pictured in the image, it is not possible to use the laser range finder data to place them. However, because it is known that the course usually run over grass, it can be safely assumed that any green pixel will correspond to the ground plane. Therefore, as long as it is possible for the camera to see green grass between obstacles, it will be possible to map that green to the ground plane. Of course, this method will not have the benefit of identifying ground features like the laser range finder. It will, however, be able to identify the ground plane in areas that the laser range finder cannot see. The algorithm will use this color-based method to create one map of the ground plane and then supplement the data with the more accurate, but potentially short range laser range finder ground plane model.

7. Ground Feature Identification in the Ground Plane

Once the ground plane is segmented from the rest of the image, it will be much easier to detect terrain such as sand pits and hazards like potholes. The image that will be analyzed will be a binary image. From preliminary tests, the color naming algorithm seems to identify

sand pits, lines, and potholes as white, so the algorithm for differentiating between the three will depend on morphological characterization of connected areas. Matlab provides powerful functions for dealing with black and white connected areas, and these will either be used directly or rewritten using custom code. Potholes and sand pits will be identified using measurements of their areas and extents in the image. Potholes can be differentiated because of their size and their round shape, which is measured by eccentricity. As each ground feature is treated and identified, it will be removed from the image to fully simplify the characterization of lines.

8. Characterization of Boundary Lines using the Hough Transform

After the characterization of the other features on the ground plane, the remaining pixels will represent both noise and the visible boundary lines. It is necessary to remove every possible object from the image so that the lines dominate the image as possible. The Hough transform is a method of finding and parameterizing straight lines in an image. It maps each point to a sinusoid in Hough space. The sinusoid corresponds to all of the possible lines that can pass through the original point. Therefore, when all of the points in an image are transformed into Hough space, sinusoids that correspond to collinear points in the original image will intersect at a point in Hough space. If there are many collinear points in the image, their Hough space sinusoids will intersect to create a maximum, as shown in **Figures 2.2.8** and **2.2.9.**

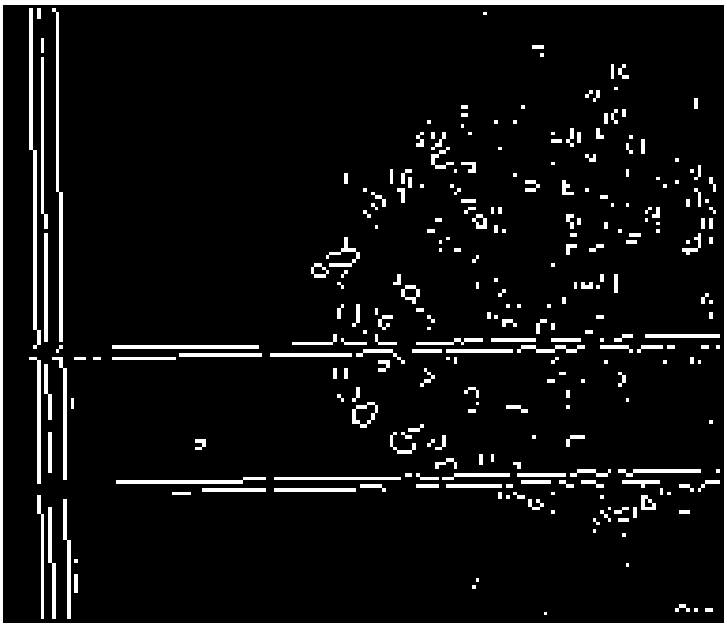


Figure 2.2.8: *Sample MATLAB Line Detection Image*

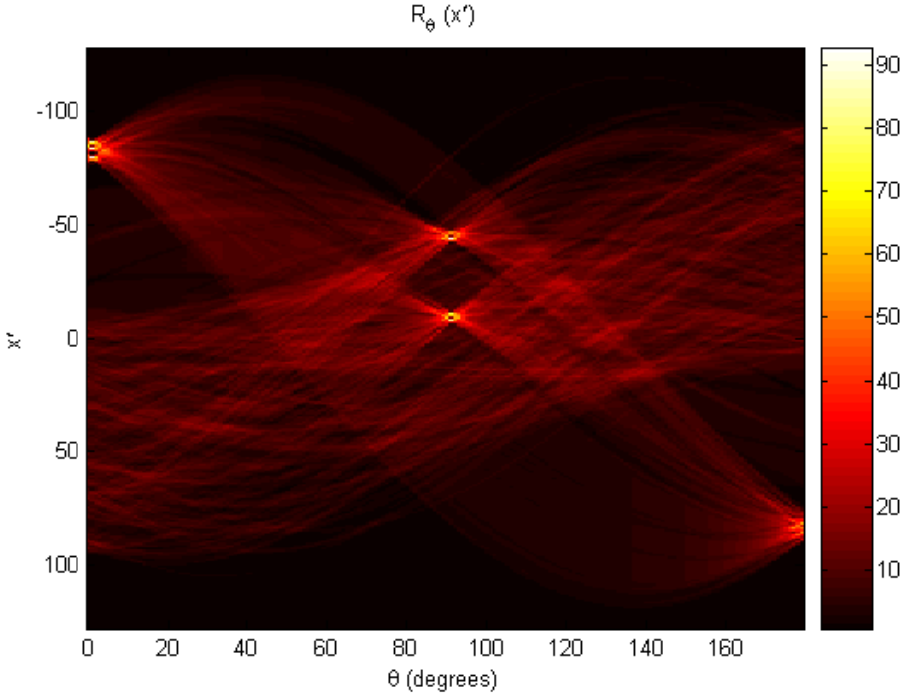


Figure 2.2.9: *Radon/Hough Transform of Figure 2.2.8*

The algorithm should create an accurate mapping of the course ahead of the robot. The direction of the course can be determined from the lines in the image. The accuracy of the algorithm decreases as the distance from the robot increases. However, placement of the obstacles should be accurate using the methods described. The idea is to create a two

dimensional overhead view that will remain constant and periodically use images captured from the camera to update the map. The heading of the robot with respect to the course can be determined using the GPS and motor encoders. Encoder information will also make it possible to match new maps with previous maps. All together, the vision algorithm is much more sophisticated than other teams that have been surveyed, including the top four teams from the 2007 competition. It should give the LSU entry an edge over the other competitors.

(2.3) Path Planning

There are three modes of operation for the real time software that executes during the course run:

- Long Distance Path Planning – Autonomous Challenge
- Short Distance Path Planning – Autonomous Challenge
- Short Distance Path Planning – Navigation Challenge

The design for the first mode is shown in **Figure 2.3.1**. This mode executes the algorithm described in the Computer Vision section. Its purpose is to generate an overhead map of the course ahead and plot a safe path through the course. This path is then passed on to the short distance mode so that it can follow the proposed path while checking for obstacles.

The algorithm begins by gathering data from the sensors through the data acquisition layer. This layer will simplify interfacing to the hardware by providing high level functions to access the desired data during program execution. Once the data has been gathered, several operations are performed. The laser range finder data is transformed into world coordinates and subsequently into image coordinates; image barrel distortion is corrected; and current heading is calculated from the GPS coordinates. Next, the laser range finder data is used to construct a ground plane map. The colors are also identified in the image.

To extend the obstacle placement capability, a color ground plane map is also created. Meanwhile, the colors in the image are segmented to remove the green grass, which is of no interest. The two ground plane maps are combined into a single map, and the ground plane is separated from the rest of the image. This results in two images: obstacles and ground plane.

These two images are passed to their respective processing algorithms. Both begin with a black and white region properties analysis. This information is used to identify A-frames on the obstacles side. Then, the obstacles' depths are estimated, and they are placed on the final obstacle map. On the ground plane, it is used to differentiate between the various types of ground features, including potholes, lines, and sand pits. Of all the ground features, the only ones of interest will be potholes and lines. Both of these features will be mapped into the final obstacle map.

While the construction of the map is taking place, the algorithm will also parameterize the lines in the image. This information will subsequently be used to extract the course direction from the image. The course direction enables a final rotation of the generated map so that the desired direction will always go from south to north in the map. The robot's position and heading will also be noted in the map.

Once the map has been rotated to account for the desired direction, the A-star path planning algorithm will execute. Once the path has been calculated, it will be quantified into GPS coordinates so that the robot will be able to place itself on the course map at a later time using the GPS data for its current position.

Figure 2.3.1: *Long Distance Path Planning*

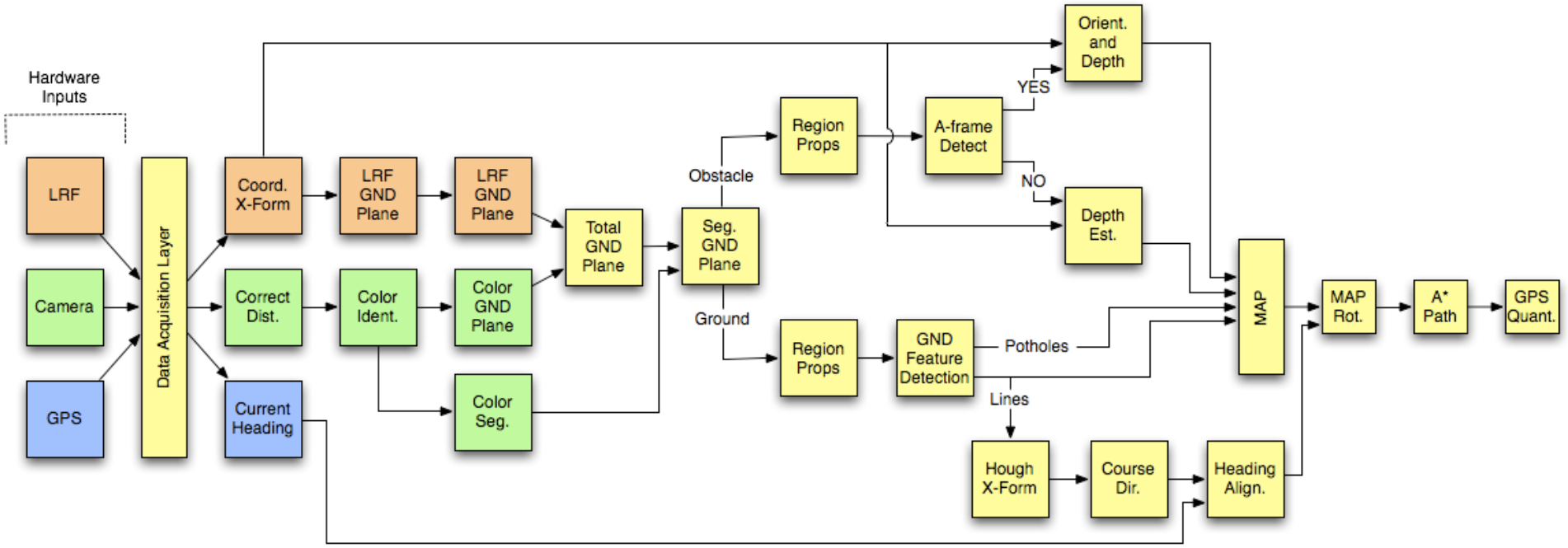
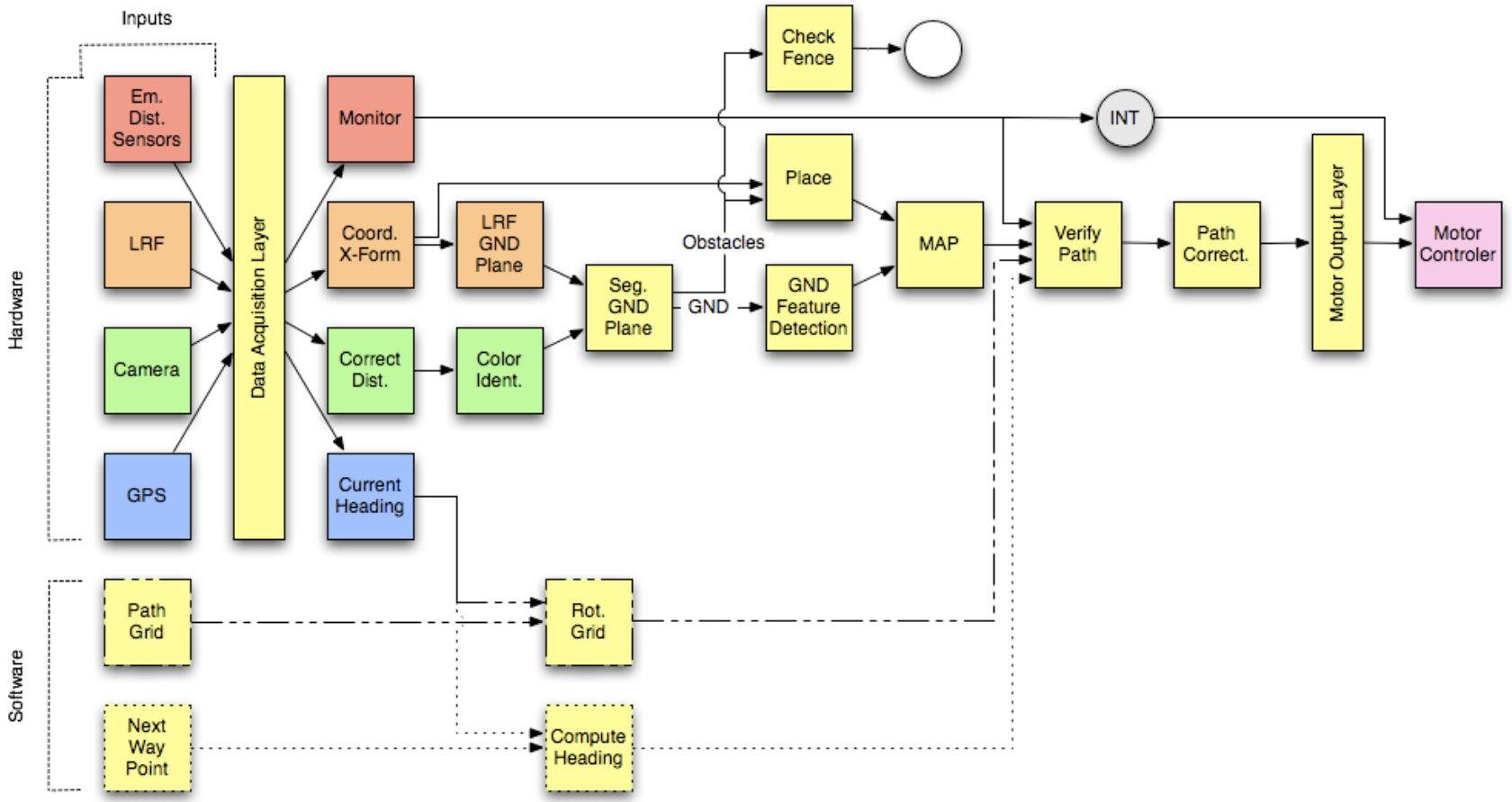


Figure 2.3.2: *Short Distance Path Planning*



The designs for both short distance path planning algorithms are combined into **Figure 2.3.2**. A significant effort has been made to overlap the two modes so that as much code reuse as possible can occur. Both implementations will be described at once.

The computer vision and laser range finder algorithm in this mode is similar to the one used in the long distance path planning, but it is simplified so that it will execute more quickly. The path planning will be concerned with short distances, so the data supplied to it will be truncated in the data acquisition layer. The extra step of calculating long distance ground plane models is removed, because the robot is not concerned with creating a long distance path. This also removes the need for the combination of the two ground plane models. The algorithm performs distortion correction on the image, identifies colors, translates laser range finder coordinates, creates a laser range finder ground plane map, and quickly moves on to segmentation of ground plane and obstacles.

Since the short distance algorithm is merely checking the pre-computed path for validity, there is no need to construct an obstacle map with the same level of complexity as the long distance path planning algorithm. Therefore, there is no depth estimation for obstacles, nor is there any boundary line parameterization. All dangerous ground features and obstacles are mapped to the obstacle grid as quickly as possible. If the robot is in navigation challenge mode, then the obstacle detection will also include a fence check. If it is determined that a fence is present, then the robot will enter a separate algorithm that will find the opening in the fence, map it as a new GPS coordinate, and recalculated the navigation path planning.

Once the local grid has been generated, the desired path must be verified. The algorithm will work with different data depending on whether the robot is in autonomous or navigation mode. During autonomous mode, the long distance path planning will have generated the desired path, and it will be passed to the short distance algorithm as a grid and

GPS coordinate set. Otherwise, the robot will be in navigation mode, and the desired path will be a function of the current position and the next waypoint to visit. The calculation will be a simple one, and a constant direction will be generated for each iteration of the algorithm. The path verification algorithm will input either of these types of data and determine if it is safe to continue on the proposed path.

If there is an obstacle in the way of the path, the algorithm will re-execute part of the long distance path planning algorithm to determine the safest path to follow. It will attempt to remain on the original path, but the presence of obstacles will take priority. The goal will be to return to the original path as quickly as possible.

Once the path correction module has executed, the result will be the path that the robot should follow. This information will be the input to the motor output layer. The motor output layer will interpret the path and generate a queue of instructions to send to the motor control hardware. This queue will be maintained in memory, and it can be interrupted at any time due to the detection of an obstacle by the emergency obstacle sensors.

The algorithm takes emergency distance sensors into account because the robot will be in motion during execution. These sensors are monitored for the presence of an obstacle. If the sensors detect an impending collision, the monitoring module will generate an interrupt for the motor control output layer so that the collision can be avoided. Also, the data concerning the location of the offending obstacle is passed to the path verification module, so that it may be used to recalculate the path.

During the execution of the short distance path planning algorithm, the computer will monitor the position of the robot. Once the robot reaches a certain point along the proposed path, the long distance path planning algorithm will execute in full alongside the short distance

path planning algorithm. This way, the robot should not have to stop to recalculate the desired path.

The fact that the short distance path following mode will execute the majority of the time reduces the urgency and time constraints for the long distance path planning. Also, the algorithm is designed to be more intelligent because of its foresight. Complicated traps will pose a reduced threat to the robot because of the overhead view.

Essentially, the path planning is modeled after the method that a human would take when tackling the problem. First, create an estimated path to follow based on available information. Then follow that path and update it or correct it as new information becomes available.

(2.4) Software Platform

The software platform that was chosen was the Linux operating system for its customizable kernel features and unlimited hardware access. Also there is access to a lot of free open source code and documentation. The Linux kernel can be totally re-compiled to allow for a smaller installation with only the required features. This frees up memory and hard drive space for other processes to use making the overall performance of the system much higher. Hardware access is a very important feature that is needed to interface with all of the external sensors used by the robot. The Linux operating system is capable of this by allowing any software that is developed to interact with its hardware counterpart seamlessly without having to worry about access restrictions. The Linux community is a big entity, which contains millions of people who support Linux software development. With this in mind, source code and documentation for many things are available freely online. This resource will reduce the time needed for software development.

(2.5) Hardware

(2.5.1) Hardware Platform

The Hardware Platform was overdesigned since there is no method in quantifying the processing power needed. The design consists of a Micro-ITX motherboard, 2 GB of DDR2 memory, a Core 2 Duo CPU, an 8000 series nVidia graphics card, and a 60 GB hard drive. The Micro-ITX motherboard was chosen for its small form factor, which is very important for placement on the robot. The 8000 series nVidia Graphics Card is needed because it is the most compatible with GPU offloading. With this type of offloading some of the image processing tasks can be ported to the GPU to allow for faster and more efficient execution. A significant amount of data will need to be processed in real-time. To account for this large amount of processing a Core 2 Duo CPU was chosen, which will allow us to multithread by processing information on two CPU cores. Because operations such as image processing are memory intensive, 2 GB of RAM was the chosen amount. The total amount of hard drive space needed is not clear at this time so a 60 GB hard drive was chosen to avoid any space issues.

(2.5.2) Laser Range Finder and Camera

The laser range finder and camera both are crucial components to the overall success of the robot. With this in mind, both components needed to meet certain requirements to perform their intended tasks successfully. These requirements are listed below:

<u>Laser Range Finder:</u>

- Viewing angle of at least 180 degrees
- Update frequency of about 8hz
- Can function outdoors
- Defined communications interface

To effectively detect and avoid objects, the laser range finder must have a 180-degree viewing angle, an 8 Hz update frequency and the possibility of outdoor usage. It will be beneficial for the robot to have the ability to detect obstacles to the sides as well as directly in front. This requires a minimum viewing angle of 180 degrees. To process data from the laser range finder while the robot is moving, an 8 Hz update frequency is needed. This means that data will need to be processed every 125 ms. Since the robot will compete outside, sunlight and humidity are both factors that need to be considered. Finally the laser range finder must be able to communicate distance and position information to the robot via a communications interface.

Camera:

- Detection frequency of at least 1hz
- Communications interface
- Can function outdoors

The image processing should achieve a 1 Hz boundary and obstacle detection frequency, thus, the camera should provide a fresh image to the algorithm at least once per second. This will give the computer enough time to process the image and will allow it to come up with a path for the robot to follow before the next image is acquired. The computer will need to acquire the images from the camera. To accomplish this task, a communications interface is used. Again, since the robot will compete outside, the camera must be able to operate in outside conditions.

Multiple solutions for both the laser range finder and camera were evaluated against these requirements. The first devices evaluated were the devices that were used in the previous design. The camera from the previous design met all of the listed requirements but the laser range finder did not. The problem with the laser range finger was that it could not perform

in an environment above 85% relative humidity. To compensate for this problem certain measures will be taken which will be discussed further in the Heat Transfer section of this document. Since both the laser range finder and camera met all of the obstacle detection requirements needed for the robot, both were chosen for the IGV.

(2.6) GPS Navigation

The GPS/Compass solution had a list of minimum specifications derived for its selection. These are listed below:

- <1m Accuracy
- Differential Correction
- Compass with .1° Accuracy
- >3 Hz Position Updates
- >5 Hz heading updates
- Standardized Communications Interface
- Water Resistivity
- 12V operation

Two of these requirements go hand in hand. An accuracy of <1m can only be achieved with differential correction. However, differential correction does not imply a <1m accuracy. The IGV must be able to enter a circle that is 2m in diameter, so a <1m accuracy was specified for the unit. The compass must also be within .1 degree of accuracy as specified in the path planning section. A 3 Hz position update would provide the following feet per update values:

Speed (mph)	Feet/Update
5	.49
4	.98
3	1.47
2	1.96
1	2.44

The GPS requirement of <1m accuracy means that anything under 3 feet/update is acceptable. In addition to position update frequency, a heading update frequency was derived for the IGV. A 90 degree per second turn was estimated to be the maximum turning speed for the robot. The only time where an increased turning radius could be accomplished is at a

standstill due to tire slippage. This tire slippage was tested on the previous robot and should not differ significantly on the new design. The degrees per update were calculated for various angular velocities and they are displayed in the following chart:

Speed (deg/sec)	Deg/Update
90	18
60	12
45	9
30	6
15	3

An 18 degree/second update rate can be reached with a 5 Hz refresh rate. The minimum refresh rate of 5 Hz was decided in the path planning section above.

The final three specifications had to do with interfacing concerns. Once again, a standardized interface is necessary, and RS232 serial is preferred. The GPS unit will be mounted on top of the robot and the competition will be allowed to continue in light rain, so the unit must be waterproof. The computer and other accessories will be running off of a 12V auxiliary battery. The GPS must be able to function on this 12V battery.

The GPS unit that is available from last year's design fits all of these specifications and it is also free. Other GPS units were researched, but most of them were in the multi-thousand dollar range and were immediately ruled out due to budget concerns.

This GPS, the CSI Wireless Vector, updates position at 5 Hz and heading at 10 Hz. It communicates with a standard NMEA 0183 protocol and has a position measurement accuracy of <1m and a heading measurement accuracy of <1m.

(2.7) Motor Controllers

The motor control solution that was chosen had to fit numerous requirements to be a perfect fit for the IGV. These requirements were separated into core requirements and special features. The core requirements are listed below:

- 45 A of Continuous Current Handling
- 24-36V Output
- Emergency Stop Input
- Hardware Limited Speed Control
- Digital Communications Interface
- Closed Loop Operation with Encoders

The motors were determined to draw up to 45 A of current at a maximum. This was determined in the power requirements section. They also run at both 24V and 36V. If more speed is needed, the voltage can be increased. In addition to power, the controller must include features specified by the rules. The rules say that the robot must have “hardware limited speed control” and an emergency stop input. The final requirements are needed for ease of interfacing. The controller must be able to communicate digitally with the computer in order to achieve precise control. Closed loop operation is also necessary for precise control of speed and distance traveled.

In addition to these core requirements, special features were specified that would make interfacing and testing easier. These features are listed below:

- RS232 Serial Support
- PID Closed Loop Operation
- Trapezoidal Motion Profile
- Joystick Controlled Operation
- Voltage, Current, Temperature Monitors
- Datalogging Output

Many of these features are not crucial for the success of the IGV, but are still useful. Also, some of these may be overcome by implementing in software, but the most limited resource is time, and coding is not conducive to time saving. The first of these requirements is an RS232 serial interface. This standardized interface is easy to write code for, and very scalable

in a PC environment. The next feature was PID closed loop operation. “This technique has a long history of usage in control systems and works on performing adjustments to the power output based on the difference measured between the desired speed (set by the user) and the actual position (captured by the tachometer).” [6] In addition to PID operation, a trapezoidal motion profile will allow for a smooth acceleration and deceleration when an input is applied to the controller. Joystick controlled operation makes testing easier and monitors are good for detection of faults and prevention of damage to the controllers. Finally, a data logging output makes logging encoder clicks easier so that we can accurately trace back a path to a certain extent.

These requirements and features were used to evaluate multiple options. The first of which was a custom motor controller design. This could accomplish all of the specifications with some extra thought in design, but the extra time in designing something so complex is overwhelming for a project of this magnitude. The second option was to purchase an off the shelf design. The first controller examined was the motor controllers that were used in the previous design. This option met all of the core requirements and included all of the optional features except for trapezoidal motion profiling. Although other designs were considered, the biggest factor in deciding upon this design was cost. The lack of trapezoidal motion profiling can be easily implemented in software therefore it was not a discerning factor. The motor controllers were available and did everything necessary for the functionality of the robot therefore they were chosen for use in the IGV.

(2.8) Chassis Design

The chassis is one of the most important parts of the design of the robot since its structural integrity ensures that the vehicle effectively participates in the competition. To satisfy the engineering specifications outlined earlier in this report, careful consideration had to be given to the structural design and material selection of the chassis. Throughout the design process, the chassis evolved from a simple metal box to an aesthetic frame that provides both equipment protection and functionality.

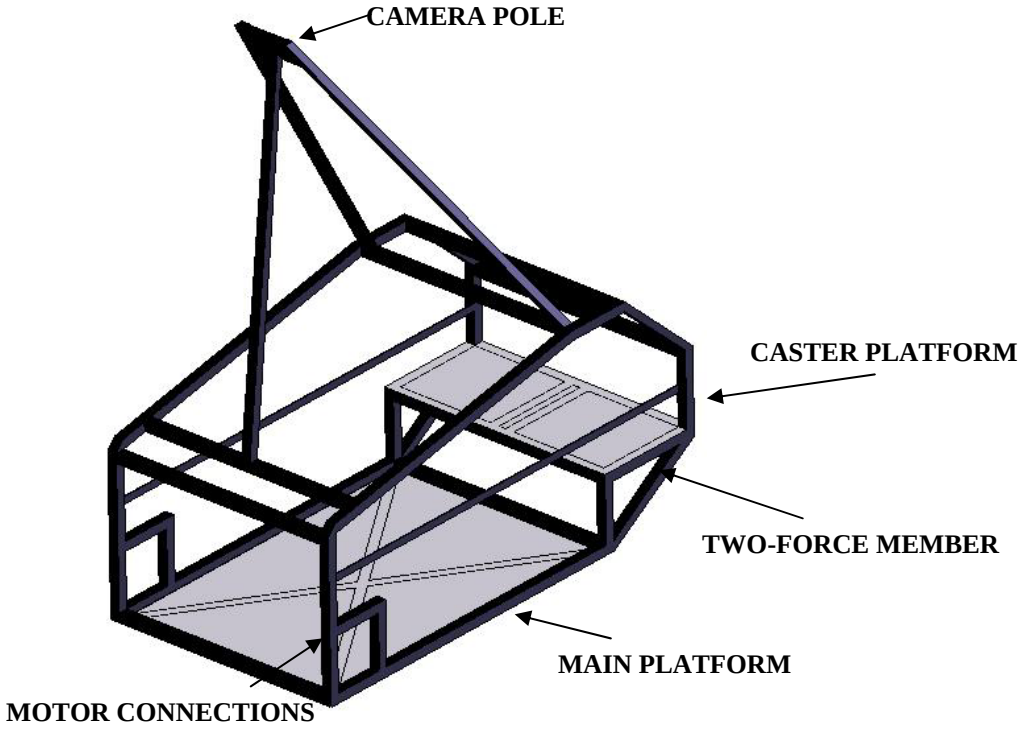


Figure 2.8.1: *Aluminum Chassis Prototype*

The chassis consists of three levels, each of which will hold different equipment. Since a low center of gravity provides increased stability, it was decided that the heavier components be placed on the bottom level. Being that the batteries and the payload are significantly heavy, they were used as the basis for the volume requirements of the chassis. The main platform

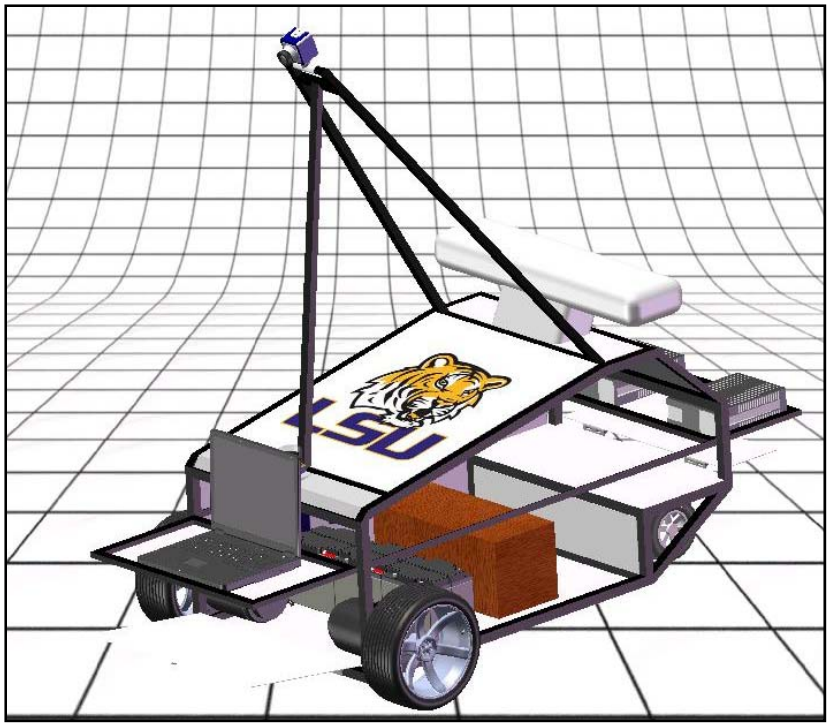
(bottom) was therefore designed to hold six batteries, the competition payload, and the two motors in the front. Also, the back portion of the vehicle was raised to allow enough space for the caster's overall height and rotational circumference. The raised platform created a critical point in the back support bars due to the increased moment transmitted from the back caster. To counteract the stress on the bars, a two-force member was added to both sides connecting the rear portion to the main platform.

The second level (shown in **Figure 2.8.2**) can slide in and out of the frame and will be used to access controllers and processing units quickly for maintenance. These platforms will be composed of the same mechanism as common drawers, which have a roller bearing mounted on a rail attached to the main frame. The placement of all components can be seen in **Figure 2.8.2**. Extra space has been left behind the payload for cables and battery chargers.

Finally, the top level of the vehicle will provide protection for the electronic equipment inside the vehicle. It is slanted at an angle to make sure that rainwater does not sit in the top and corrodes the frame or cause any leakage. Also, the flat portion of the frame will provide a section to mount the GPS system on the top of the vehicle.

(2.8.1) Chassis Material Selection

Aluminum AISI 6061, one inch square tubing (1/16 inch thick) will be used in the manufacturing of the chassis. This material provides a combination of high strength at low weight. The 6000 series was chosen since it is readily available in the required size by one of the current team's sponsors (Metals Depot: 15% discount on all metal part purchases). Also, the 6000 series can be welded with ease and can be heat-treated for increased strength.



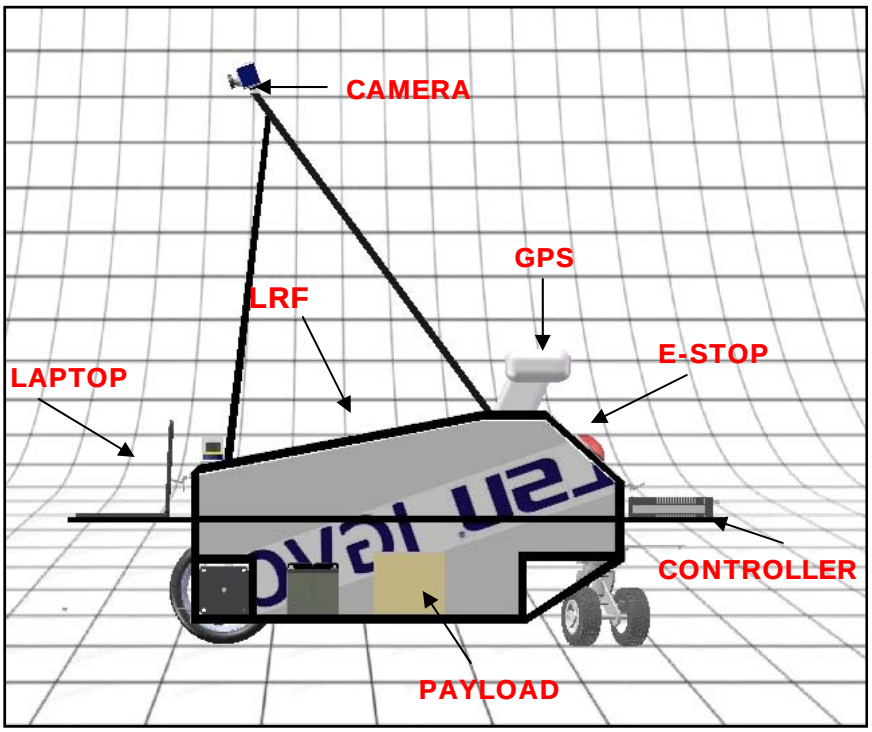


Figure 2.8.2: *Component configuration on vehicle*

Aluminum was chosen over plain carbon steel since a lower weight will result in reduced power requirements and will make the vehicle easier to transport. SolidWorks was used to determine the weight reduction from 304 AISI Steel (used by previous LSU IGVC team) to aluminum AISI 6061. The weight of the designed chassis was calculated to be 29.2 lbs using aluminum and 89.3 lbs using steel. This corresponds to a 67% weight reduction with the use of the new material.

Due to the lower yield strength of aluminum when compared to steel, an extensive stress analysis was performed on the vehicle frame to ensure that the chosen material will be able to support the loads with an acceptable factor of safety.

(2.8.2) Finite Element Analysis- Stress Distribution

Several loading models were tested in CosmoWorks to ensure that the frame will not fail during operation. An example of a model is shown in **Figure 2.8.3** (Stress distribution plot). The loads modeled on this example were 1000 lbs distributed load on the center metal sheet and 800 lbs force transmitted by the back caster onto the rear portion of the frame. This loading condition is around 4 times larger than actual operating loads. As it can be seen in the von Mises stress distribution plot, the highest stress occurs on the joints on top of the motors. Similar high stress points can be seen in other support bar joints around the vehicle. The points of high stress have a magnitude of 0.86 ksi, which is well below the yield strength of aluminum (7.99 ksi). These results show that the vehicle will be far from failure during operations. On the other hand, care must be taken during the manufacturing process since high stress points occur around weld joints. The manufacturing section of this report will discuss different techniques that can be used to reduce the possibility of failure in these areas.

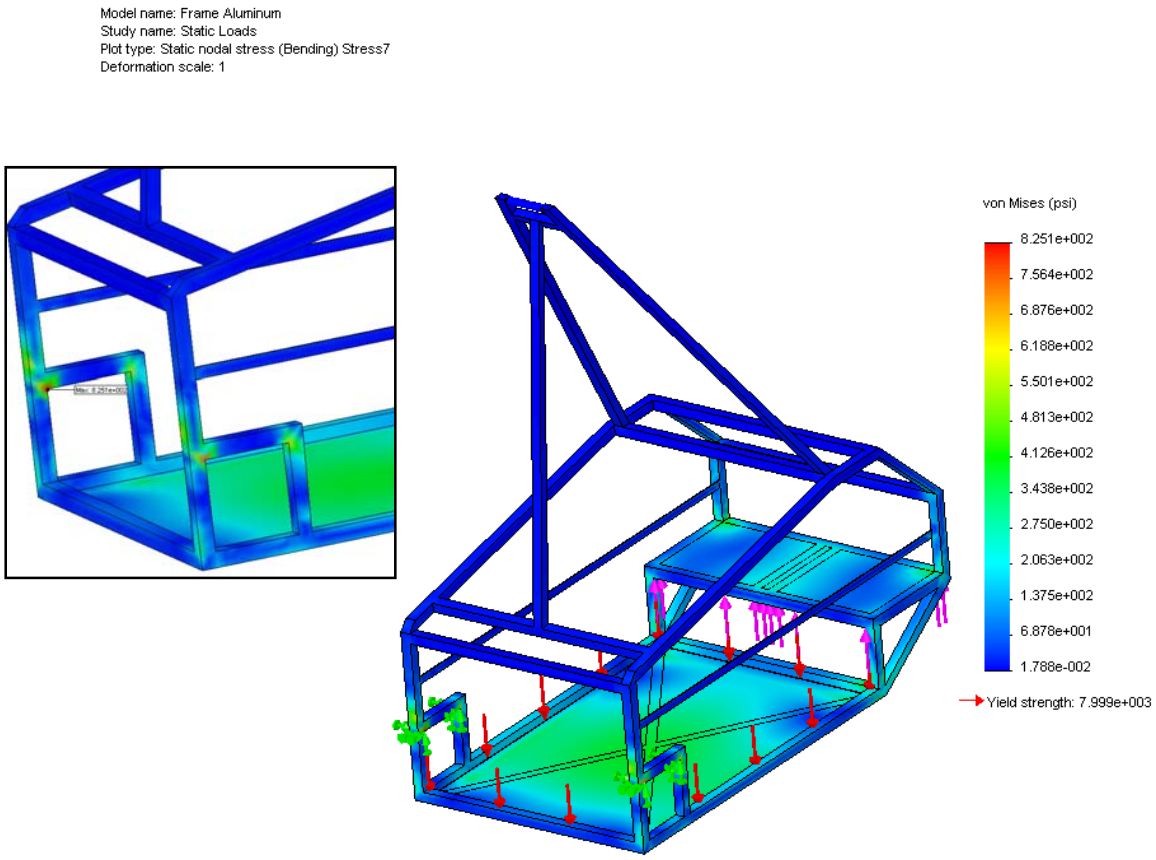


Figure 2.8.3: *Von Mises Stress Distribution Model for Chassis*

(2.8.3) Finite Element Analysis- Displacement

The same loading conditions described earlier were used to determine the displacement of different points in the chassis. The URES displacement plot shown in Figure 2.8.4 provides a graphical representation of the movement of the chassis. It can be seen that the highest displacement occurs at the top of the camera pole. Although movement at this point could cause problems during image data acquisition, the magnitude of the displacement is so small (~1mm) that no effect on the camera is expected.

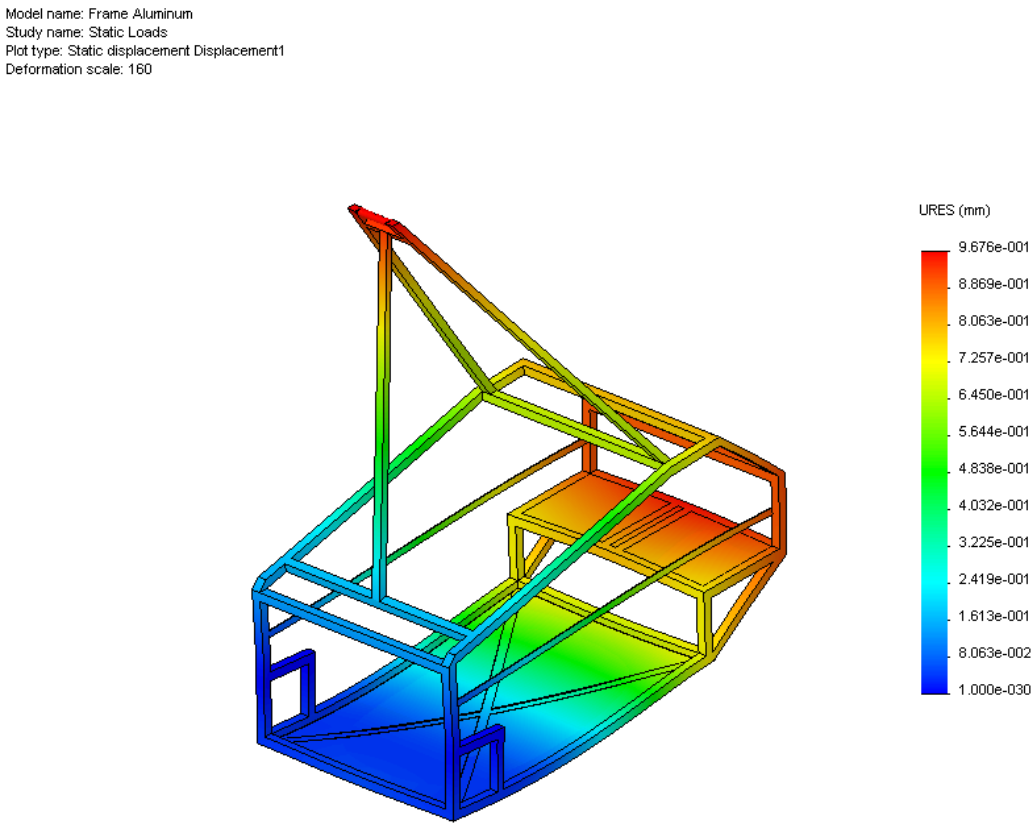


Figure 2.8.4: *URES total displacement of Chassis*

(2.8.4) Finite Element Analysis- Natural Frequencies

Another mechanical effect that could consequent in failure of the chassis is cyclic loading. This is a concern in case the vehicle chassis is exposed to vibrations at a frequency close to the natural frequency of the aluminum frame. The effects of natural frequency vibrations can be observed in **Figure 2.8.5.** Again, CosmoWorks FEA was used to determine the natural frequency for different loading modes. The vibration frequencies from this analysis can be compared to the vibration testing analysis that will be performed after the manufacturing of the frame. The team will need to ensure that proper vibration damping is in place to prevent the frame from approaching any of the natural frequencies outlined in **Table 2.8.1**.

Mode No.	Frequency (Hertz)
1	20.70
2	25.27
3	64.66
4	74.91
5	83.45

Table 2.8.1: *Natural Frequencies of the Chassis under Different Loading Modes*

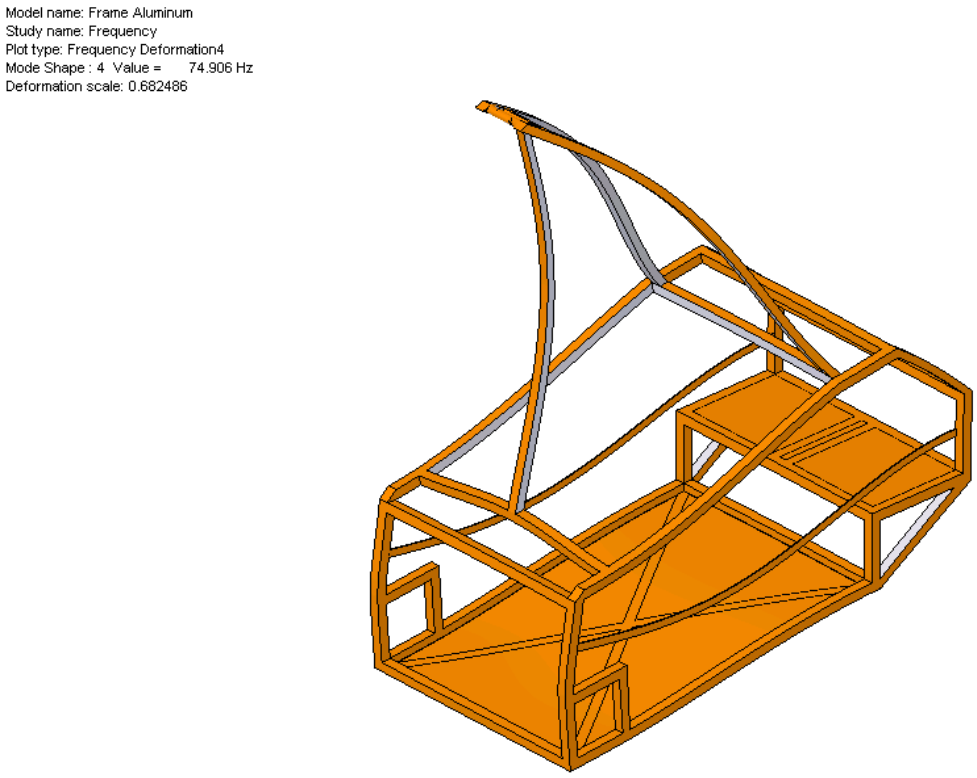


Figure 2.8.5: *Shape of Displaced Chassis when exposed to a natural frequency vibration*

(2.8.5) Weatherproofing

The frame of the vehicle will be covered with 20 gage (0.036 inches thick) cold rolled steel sheets on all side openings of the vehicle. This was chosen over plastic sheets due to increased strength, durability and lower cost. The sheets will be bolted onto the frame and sealed with rubber or silicone sealant around the gaps to prevent from water leaking inside. This concept will also provide for a more aesthetical design of the overall robot. Additionally, the sheets do not only have to be attached during adverse condition but can remain on the body at all times. Finally, the sheets will need to be painted to protect them from corrosion.

(2.9) Motor Sizing

The condition where the vehicle will need the highest amount of power is when it will be going up the 15° incline in the competition. The desired travel speed thru the incline is 5 mph (limited by competition rules). Using dynamic equations, a relation between the weight of the vehicle and the horsepower required can be found. By placing the previous team's vehicle on an increasing incline and measuring the angle at which motion started, the coefficient of rolling friction was estimated to be around 0.06. **Figure 2.9.1** shows the results of the dynamic calculations (Refer to Appendix F).

The estimated final weight of the vehicle (calculated in SolidWorks) is around 110 kg (243 lbs), which will therefore require a total HP of around 1.0. The motors chosen to provide this amount of power are two NPC-64038 4-pole motors weighing 13 lbs each. They are each rated at a maximum of 24V, 230 rpm and 1.3 HP with a 20:1 gear reduction. Also, the stall torque for each motors is 68.75 ft lbs.

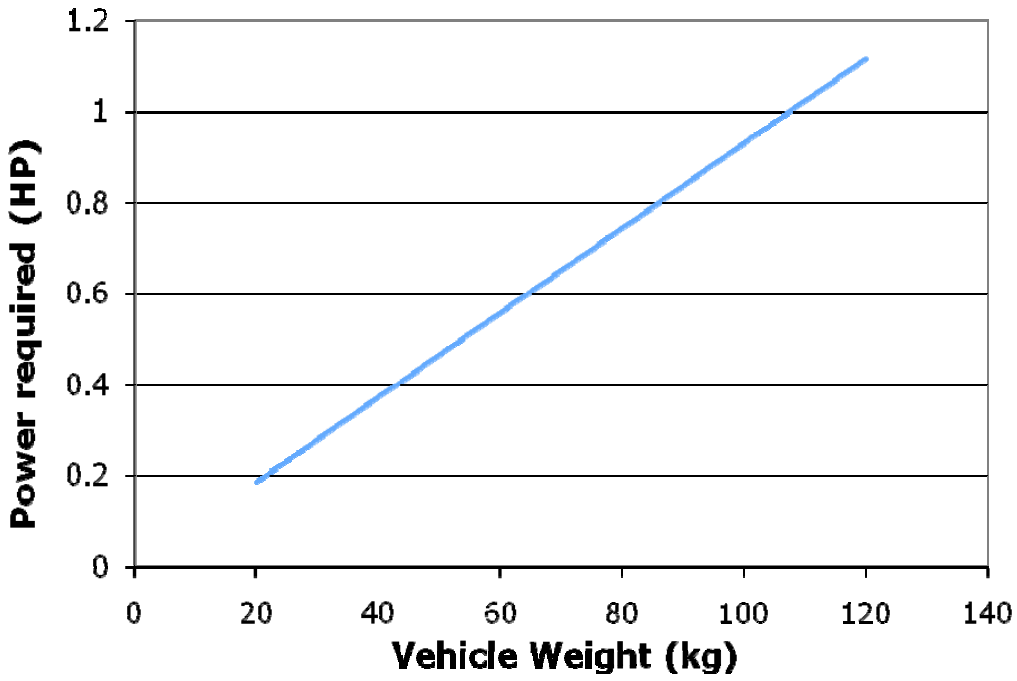


Figure 2.9.1: Maximum *HP required vs. vehicle weight*

Torque calculations where performed to ensure that the torque required to move the vehicle would not exceed the stall torque. For the 12-inch diameter tires chosen (Refer to Speed Control Section 2.10) the total torque to maintain a speed of 5 mph up a 15° incline was calculated to be 38.4 ft lbs. Also, the torque required to begin motion from rest is 14.8 ft lbs on level ground and 52.1 ft lbs at a 15° incline. All calculated values are well below the net torque that the motors can provide (137.5 ft lbs).

The output power of the motors was plotted from manufacturer's dynamometer data versus the amount of current needed. It can be seen that at 25 amps, one motor can provide 0.5 HP for a total vehicle power of 1.0 HP. As expected, higher current inputs provide more power output. This means that the purchased motors can provide more than enough required power for the operation of the vehicle.

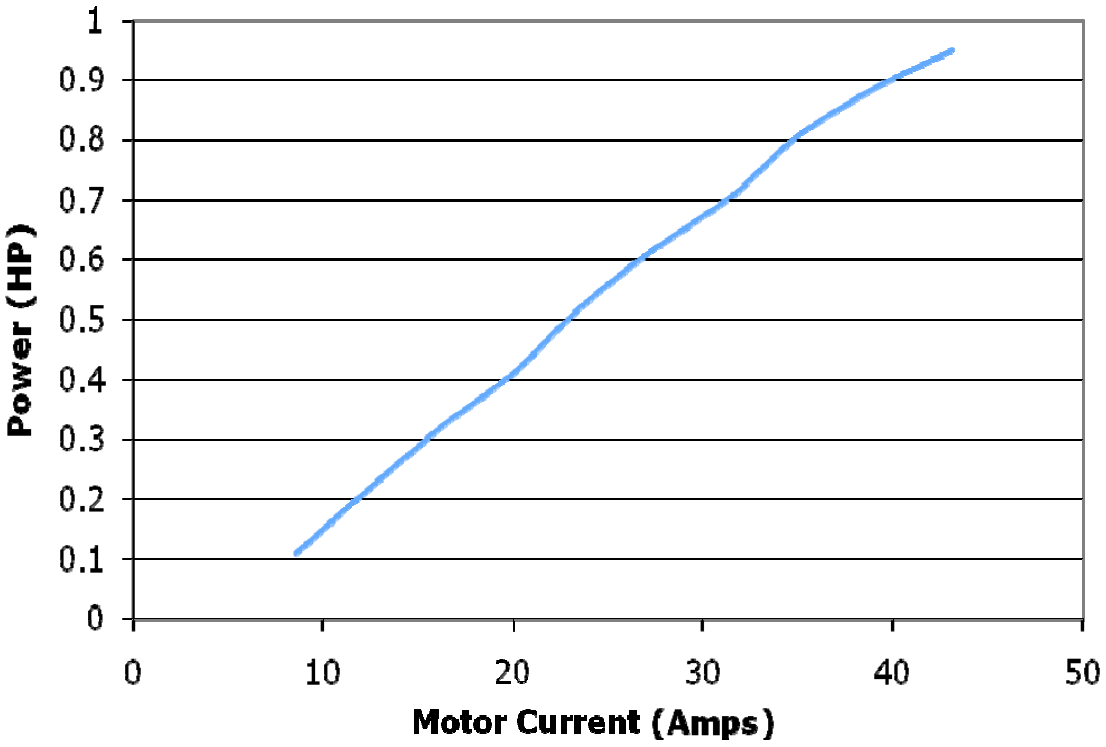
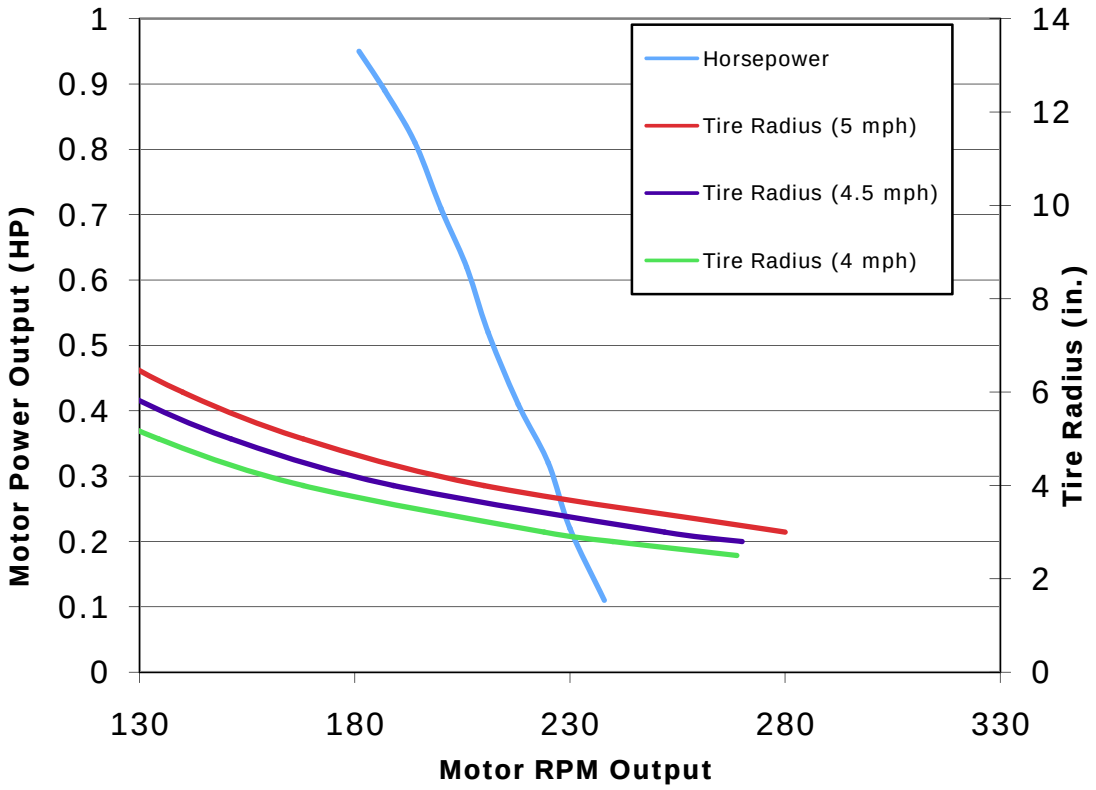


Figure 2.9.2: *Horsepower vs. Motor Current*

(2.10) Speed Control

The manufacturer's data of the purchased motors provided information of the output speed at different power outputs. This means that if no gear reduction were used, the shaft would rotate the wheels at the same speed as the motor rotation. From dynamics, it is evident that larger tires have a higher outer point velocity than small tires. Using relations for angular speed to translational speed, a plot of required tire radius for different RPM at different vehicle speeds was created. Also, the manufacturer data of the motor speed was plotted in a different axis.



The tires chosen for the vehicle are 6 inches in radius since this type is readily available in pneumatic tires from manufacturers like McMaster-Carr and The Robot Marketplace. The required motor RPM for 5 mph can therefore be found from **Figure 2.10.1** and is equal to a value of 140 RPM. This value can be programmed into the motor controller and will therefore limit the robot to the speed required by the competition.

(2.11) Heat Transfer & Humidity Control

The manufacturer of the laser range finder specifies that it will work in environments with 85% relative humidity or less. This limited humidity will be achieved through the addition of sensible heat. In the worst-case scenario, air bordering saturation, it can be shown that a temperature increase of 7° F would be sufficient to decrease the relative humidity below 85%. This is shown on the Psychometric Chart figure to the right, in which the blue line represents 85% relative humidity. It is thus desirable to increase the air temperature surrounding the laser range finder 7° F above the ambient. The heat required to achieve this temperature increase can be found using a conservation of mass and energy. Simplifying the results yields the following relationship between the dissipated heat load (h_{sens}), the air flow rate in cubic feet per minute (cfm) and the temperature increase (ΔT) obtained in the space.

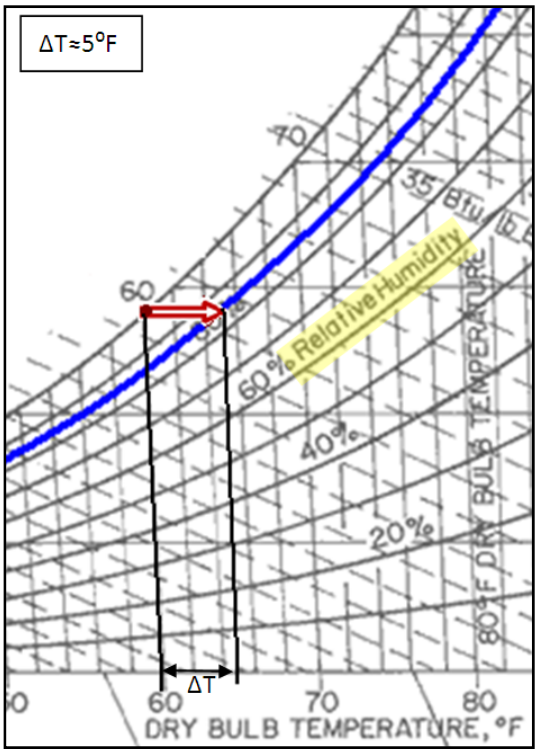


Figure 2.11.1: *Psychometric Analysis*

$$\Delta T = \frac{h_{sens}}{1.08 * cfm}$$

The heat dissipation of the electrical equipment used will vary depending on usage but is estimated to be in the range of 30-50 Watts. The following figure shows a plot of the required air flow rate to achieve a desired temperature gradient as a function of heat dissipation.

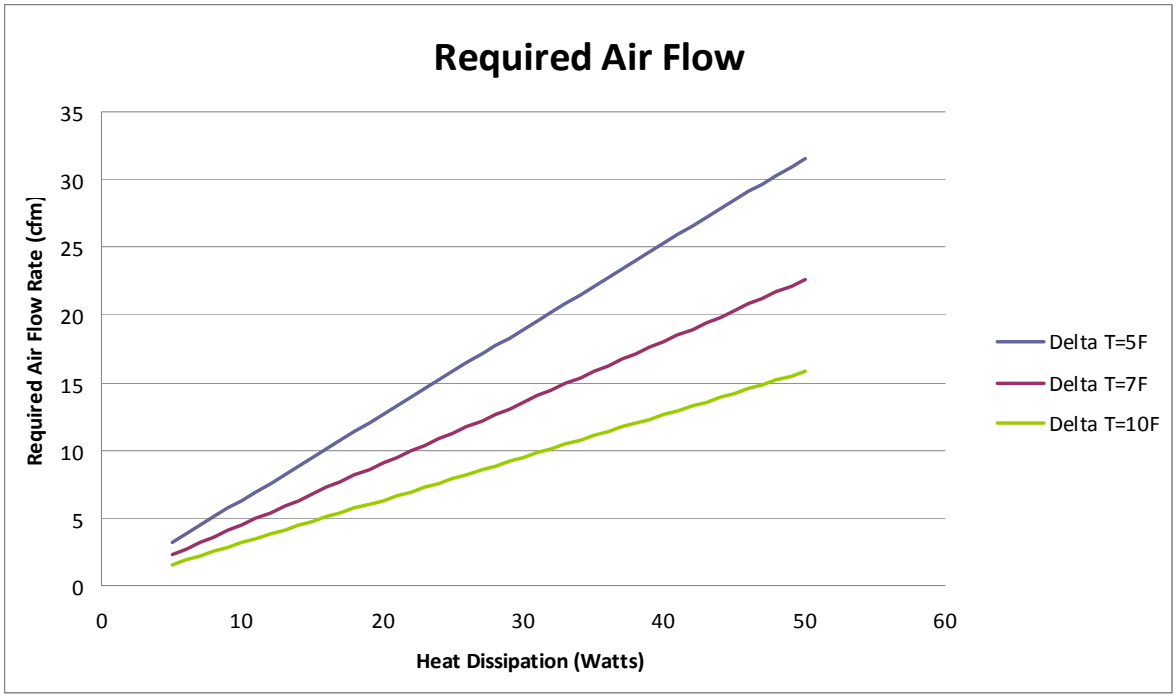


Figure 2.11.2: *Required Air Flow*

It is also important to estimate the transient heating process. The following plot shows the time required to reach a given temperature increase assuming a constant volume.

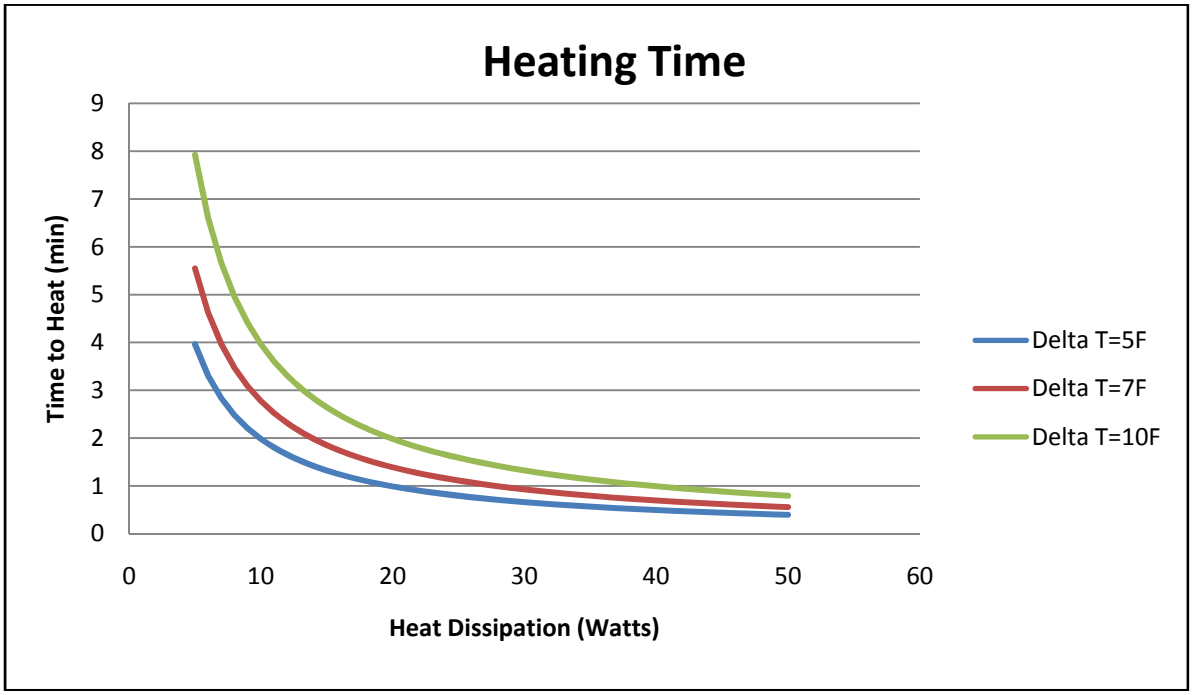


Figure 2.11.3: *Heating Time*

Even at a low heat dissipation of 10 Watts, the desired 7˚F temperature increase can be achieved in less than 5 minutes, after which the constant flow of air should maintain the temperature gradient. The computer and video card will likely create the most heat and so they will be placed directly under the laser rangefinder to achieve the dehumidification quickly. The exhaust fan will be mounted at the top of one of the side panels in the rear of the robot. Slots will be cut in the opposite side panel near the front of the robot. This setup provides exhaust of the hottest air first and supply of cooling air to the critical electrical components. The rangefinder will also be protected from the external elements on top and sides by a glass container. Tests have proven that the laser image is not affected by a glass enclosure. Further testing is required for accurate design of the exhaust system and the details of the testing plans can be found in the following sections.

(2.12) Stability

Static stability is maintained as long as the center of mass is located within the triangle made from connecting the x and y components of the three wheels as shown below:

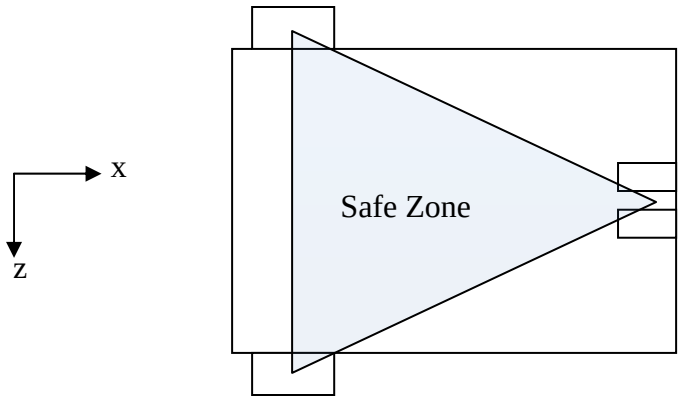


Figure 2.12.1: *Static Stability*

For this reason the center of mass was deliberately positioned closer to the front of the robot . When the robot is on an incline this “Safe Zone” will decrease in size as the x-components of the wheels become closer together. In this case, the y-component becomes a factor. The y-component also affects the dynamic stability during turns and deceleration as shown in the following sections.

Since the robot is front wheel drive, forward acceleration is not a stability issue, but deceleration will cause a moment which can potentially tip the robot. The critical deceleration occurs when the rear wheels loose contact with the ground and the normal force goes to zero as shown in the following figure.

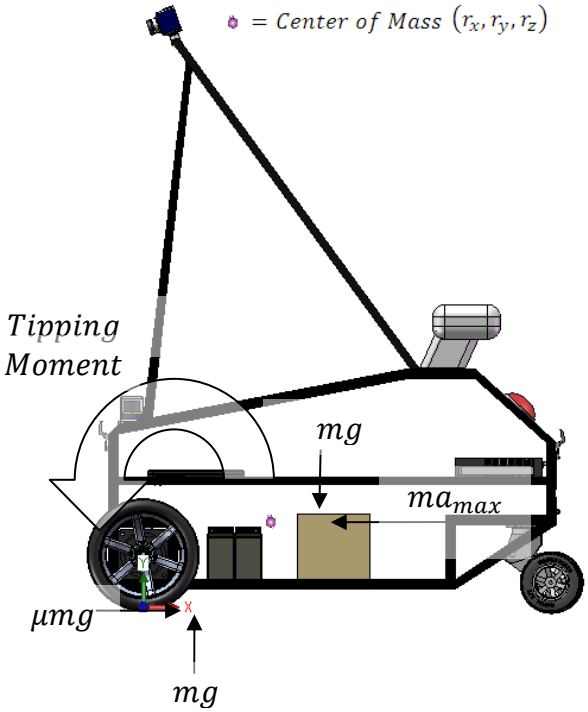


Figure 2.12.2: *Deceleration Stability*

The center of mass was positioned as low as possible in an attempt to lower the tipping moment. This involves placement of the heaviest items (batteries & motors) on lowest level of the frame. The precise location of the center of mass was calculated in detail using SolidWorks

by inputting actual dimensions and weights for each individual component. The resulting center of mass in Cartesian coordinates (x,y,z) is (17.69", 7.26", 12.78).

By summing the forces and moments for this condition, the maximum acceleration can be found. The stop time is then calculated assuming the robot will stop from a speed of 5mph to 0mph. The results are as follows:

$$a_{max} = -\frac{r_x}{r_y} g = 23.9 \frac{m}{s^2}$$

$$Min.\ Stop\ Time = 0.0935s$$

The competition rules specify that inclines with a maximum slope of 15° must be traversed in the obstacle course. Any moment created from rapid deceleration will be amplified if the robot is in the process of descending an incline. In essence, the r_x and r_y in the previous equations are changed to new values x and y which represent the horizontal and vertical distance from the front wheel to the center of mass. For a 15° incline, the following pair of equations can be found.

$$tan(15°) = \frac{x}{y}$$

$$r_x{}^2 + r_y{}^2 = x^2 + y^2$$

Solving the equations yields x=4.95" and y=18.49". Substituting into the previously mentioned equation, the true maximum acceleration and minimum stopping time are found to be.

$$a_{max} = -\frac{x}{y} g = 2.626 \frac{m}{s^2}$$

$$Min.\ Stop\ Time = \mathbf{1.18}\boldsymbol{s}$$

In the interest of minimizing processing time and maintaining simplicity, it is beneficial for the robot to steer via differential motor speeds. For this setup it is easy to achieve a 0° turning radius. Though this can be beneficial, it can produce a moment which could tip the robot. The critical turning radius can be shown to be a function of the center of mass, velocity,

and wheel location. During a constant velocity turn there are four characteristic forces acting on the robot: normal force on all four wheels, friction force on the two rigid front wheels, gravitational force through the center of mass and centrifugal force through the center of mass. The critical point occurs when inside wheel losses contact with the ground and the normal and friction force are zero as shown in the following figure.

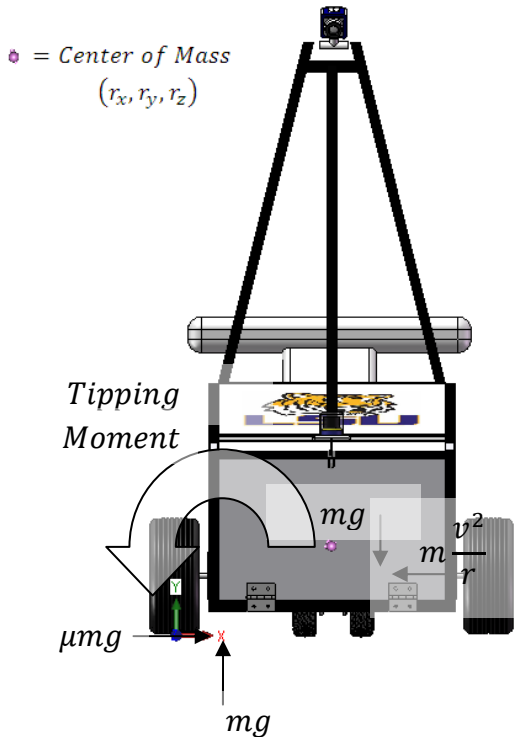


Figure 2.12.3: *Turning Stability*

For this loading, the forces and moments can summed to achieve the following relationship where w and l are the width and length of the robot, v is the velocity, g is the gravitational acceleration, and r_x, r_y, and r_z are the components of the center of mass relative to the front right tire.

$$r_{min} = \frac{v^2}{\frac{g}{r_y}\left(r_x - \frac{r_z w}{2l}\right)}$$

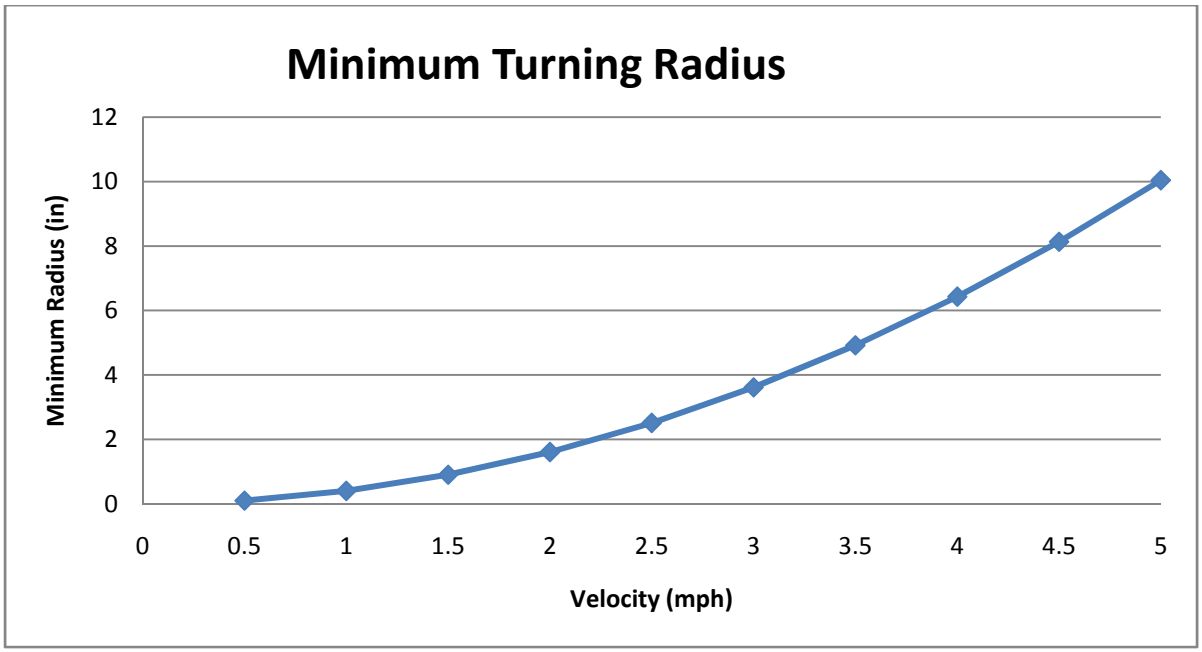


Figure 2.12.3: *Minimum Turning Radius*

Another important aspect to consider in dynamic stability is sliding of the wheels due to the centripetal acceleration in a turn. The frictional force on the wheels is set equal to the centripetal force in order to calculate the minimum turning radius. The resulting equation below is a function of the friction coefficient (μ) and the velocity (v).

$$r_{min} = \frac{v^2}{\mu g}$$

The value of the friction coefficient will vary greatly depending on the two surfaces in contact but should be in the range of 0.25 to 0.9 for the expected conditions. The following plot shows the relationship of minimum turning radius to friction coefficient for a 5mph turn.

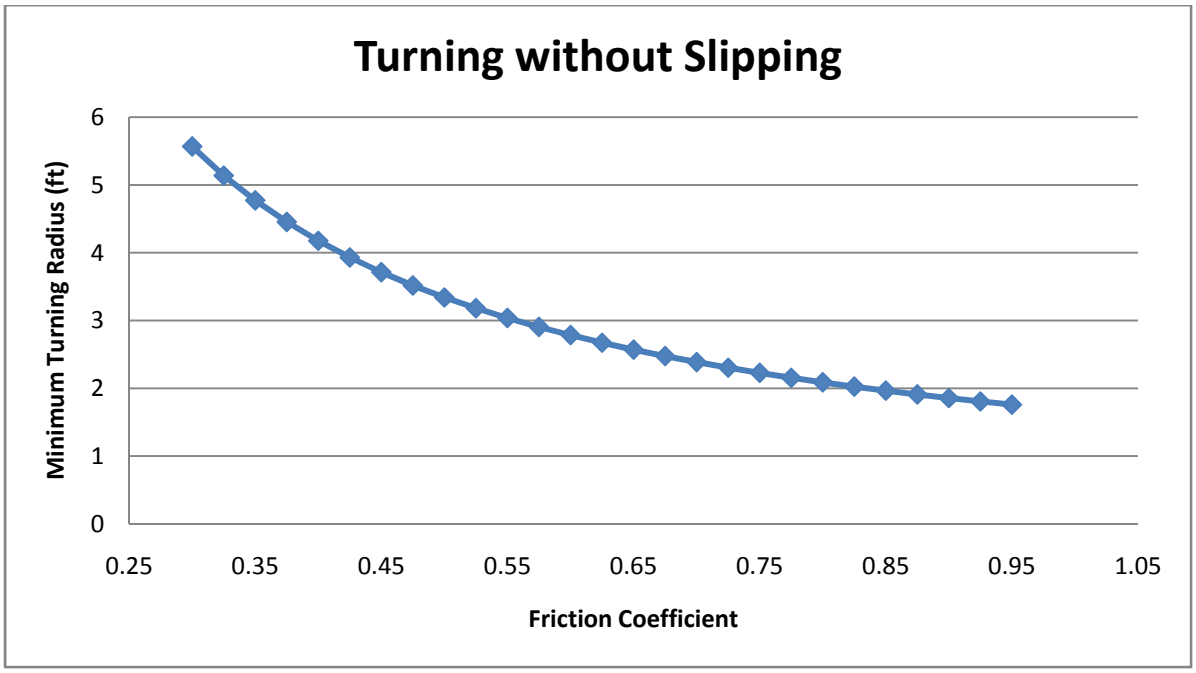


Figure 2.12.4: *Minimum Turning Radius for No Slip Condition*

When considering slipping due to the radial acceleration of the wheels the analysis is a little more complex.

(2.13) Vibrations

Vibrations were an important concern when designing the chassis, mostly because the acquired image needs to be clear. An effort was made to quantify the allowable vibration so that it could be taken into account in the chassis design.

The analysis started with the assumption that a movement of half of a pixel was the maximum acceptable displacement per image. Such vibration would introduce some element of image blur, but it would either be negligible or easily correctable with a Weiner filter.

The main focus of the analysis was finding the area represented by one pixel of the image. If the camera is angled downwards towards the ground, then the closest point on the

ground will represent the least amount of data per pixel. This corresponds to the area closest to the robot itself.

There should be no wasted area in the field of view of the camera, so the image should definitely not contain the front end of the robot. However, it is still desirable to see as far the down the course as possible without sacrificing the close range view. In any case, the worst case scenario will be if the edge of the field of view of the camera comes just to the front edge of the robot.

Using similar triangles, an expression for the area of ground expressed in the closest pixel was determined. It is shown in equation (4).

$$d_p = h_c\left[\tan(\vartheta_2 + \vartheta_3) - \tan(\vartheta_2)\right] \qquad (4)$$

This is twice the acceptable movement in the x direction. The vertical height is related to the horizontal distance by equation (5)

$$d_{horizontal} = h_c \tan(\vartheta_2) \qquad (5)$$

Therefore the maximum distance that can be moved per frame is given by equation (6).

$$d_{\max} = h_c \frac{\tan(\vartheta_2 + \vartheta_3) - \tan(\vartheta_2)}{2\tan(\vartheta_2)} \qquad (6)$$

Taking the minimum shutter speed of the camera (1/30 s) the maximum velocity of the vibration is given in equation (7).

$$v_{\max} = 15h_c \frac{\tan(\vartheta_2 + \vartheta_3) - \tan(\vartheta_2)}{\tan(\vartheta_2)} \qquad (7)$$

Using a SolidWorks model of the proposed chassis, the maximum allowable vibration speed was found to be:

$$V_{\max} = 22.95\frac{in}{\sec}$$

Preliminary tests of the 2006 model provided quantitative vibration information for several common terrains. The results are presented in the following table.

Terrain	Vibration (in/s)
Concrete/Pavement	1-2
Gravel	4-6
Potholes/Bumps	8-10

It is important to note that during testing the accelerometer was placed on the top surface of a horizontal frame member because the camera mount is longer intact. The camera mount for the 2006 robot consisted of a single vertical pole welded at the base to the top of a horizontal brace. A combination of subpar welds, poor design and inadequate pipe sizing resulted in a camera mount that allowed significant movement of the top of the pole, and eventually failed at the weld. The 2007 design will incorporate a tripod camera mount that will attempt to create a rigid extension of the frame to which the camera assembly can be mounted. In an attempt to quantify the benefit of the tripod design as compared with a single vertical pole, a displacement analysis was completed using Cosmos FEA in SolidWorks. The loading conditions and the camera mount orientations are shown below.

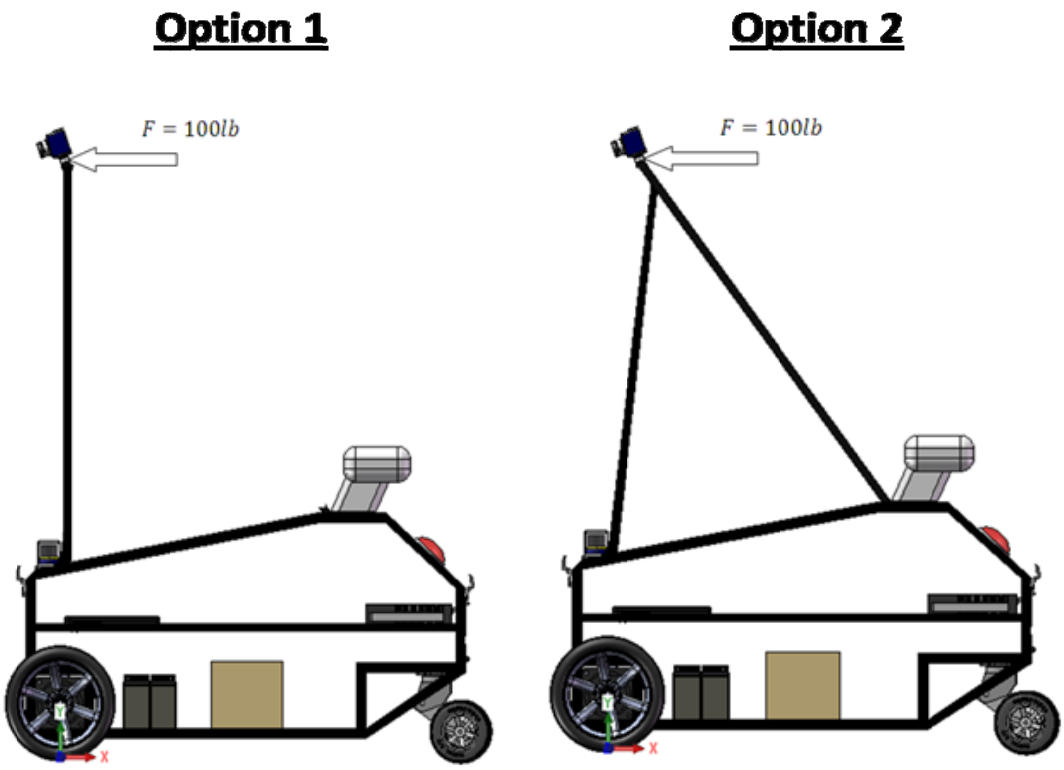


Figure 2.13.1: *Camera MountOptions*

For the loading conditions shown, the calculated displacement of the camera in option 1 is 8.21mm, as opposed to only 3.21mm in option 2. This corresponds to a 58.6% reduction in camera movement and thus a worthy design. Since the camera mount will provide a more rigid connection to the frame, vibrations induced by the ground must be reduced before they reach the frame.

During the vibration tests, the rear casters appeared to be a significant source of vibration. The casters used on the 2006 model are made of hard plastic which has worn considerably. The casters are attached to steel plates welded in a cantilever fashion to the rear of the frame. The casters are not designed for outdoor use and as a result, they would often lock in position during turns. The driven tires appear to be pneumatic but under more

investigation it can be seen that they have no air valve. It is assumed that the tires are semi-pneumatic run flat tires, which absorb little shock.

The shock absorption capability of the driven tires will be increased by using fully pneumatic tires. A 12" tire will be used because it can easily be fit to a 6" wheel rim, which is the smallest rim with readily available hub connections. Larger tires would require higher torque and result in additional power usage. Tire pressure will depend on the result of testing to be completed next semester. This testing will try to correlate the spring rate of the given tire to the air pressure inside the tire.

The shock absorption of the rear wheels is designed to match that of the front wheels. For this reason pneumatic tires will be used in place of the hard plastic. This requires the use of larger wheel diameters, which in turn requires an offset of the wheel from the caster assembly. To balance the caster two wheels will be attached to a single axle on the arm of a caster. It is hypothesized that the smaller caster tires will not absorb force as well as the larger tires, therefore a spring loaded caster design was selected for the base of the vehicle's modifications. Two internal springs absorb force as they allow movement of the caster arm with respect to the bearing and mounting bracket. The stock springs can easily be removed and new springs can be chosen to more closely match the characteristics of the front tires. The following is a schematic of the caster design, in which the springs are modeled here as a dash pot.

Figure 2.13.2: *Rear Caster*

(3) MANUFACTURING & TESTING PLANS

The following presents a description of each stage of the manufacturing and testing portion of the design process. Progress has been made in some stages but the majority will be completed in the Spring semester. Immediately following the descriptions is a list of tentative completion dates.

Stage 1. Moving Chassis (Expected Completion: 2/15/07)

We decided that a moving chassis is the most fundamental of all the measures of success. Without a moving chassis, no further objectives can be met. This stage includes full construction of the chassis with the motors mounted, the motor controllers installed, and the wheels attached. At this point, the 2006 vehicle has been reconstructed to use in Stage 2.

The 2007 chassis construction will be completed in three steps. First all material will be ordered and joints will be dimensioned and designed. Second, practice-welding sessions will be held until welding confidence is achieved. Meanwhile tests will be completed on the tires and caster springs in order to correctly model the spring rates. Finally, the frame will be manufactured and the drive components attached.

Stage 2. Working Software-Based Motor Control (Expected Completion: 1/18/07)

This stage involves obtaining working software-based motor control. In essence the robot's motion can be fully controlled at all time at the completion of this stage. This includes interfacing the motor encoders with the motor controller, tuning the motor profile for smooth operation, and having a working software frontend on the pc. This stage has already begun using the 2006 robot and it will be completed upon finalization of the motor control module.

Stage 3. Basic Sensor Interfacing (Expected Completion: 2/01/07)

Successful interfacing of basic sensors is the next required milestone. The sensors give crucial feedback to the robot allowing it to decide upon a certain course of action. This step includes integrating a computer vision solution via hardware and software, measuring distances from some form of distance sensor to locate obstacles, implementing the GPS transceiver and acquiring a position from it. The focus is connectivity of the components and initial testing of functionality and not the full processing algorithms. Effects of vibrations on the camera and the computer vision will also be addressed. This stage is nearly complete. Interfacing of the GPS is the only remaining task.

Stage 4. Self-awareness (Expected Completion: 3/07/07)

The next stage is complete when the robot is self-aware of its current position, speed, and heading based on sensor data. The robot must obtain this information before it can know to adjust its position and speed. This stage has two aspects. The first is to calculate actual speed based on encoder data, motor gear ratio, and tire size. The second is to calculate position and heading information from the GPS unit.

Stage 5. Obstacle, Line Detection and Avoidance (Expected Completion: 3/14/07)

The fifth measure of success is obstacle and line detection. This takes the self-awareness of the robot a step farther, requiring the robot to stay on a set course while avoiding obstacles in its path. This step involves integrating the distance sensor and computer vision solution to work in unison to detect the exact location of obstacles. This step also involves using the computer vision solution to detect course boundaries

and potholes. Basic algorithms will be programmed to provide obstacle avoidance or line following.

Stage 6. Emergency Obstacle Avoidance (Expected Completion: 3/21/07)

The next step involves the implementation of a back up obstacle avoidance system. The emergency obstacle sensors will be installed on the robot and tested. A subsystem will be put in place and finalized so that it can interrupt the system and issue a stop command to the motor controller.

Stage 7. Intelligent Path Navigation (Expected Completion: 4/25/07)

The seventh stage is intelligent path navigation. The obstacle avoidance algorithms will be taken a step farther to maintain optimal direction at all times, along with choosing an optimal path around obstacles. The algorithms will also be adjusted for testing of the navigation challenge, and an optimal path algorithm will be added.

Stage 8. Complete and Final Debug (Expected Completion: 5/30/07)

By this point, every piece of the robot should be functioning according to the design. The last stage is final testing and debugging. The objective is to test the function robot in as many different situations as possible.

(4) SAFETY ISSUES

Proper consideration of safety is key factor in the design of this year's robot. The 250lb metal vehicle is capable of maximum speeds approaching 10mph and once finished it will be completely autonomous. For this reason multiple layers of redundancy have been designed to ensure safe operation of the robot. These layers are as follows: speed limitation, obstacle

avoidance algorithms, emergency distance sensors, and finally mechanical and wireless emergency stop.

The speed of the robot will be hardware limited by the motor controller at 5 mph during competition events. However, the speed limitation will be reduced during any tests conducted within the direct vicinity of people or property that could be damaged. The speed limitation reduces the risk of collision and it also serves as a means to reduce the severity of any potential collisions. The computer vision algorithms for detecting obstacles with the camera and laser rangefinder in combination with the obstacle avoidance algorithms will serve as the primary means of preventing collision. In the event that this system fails to avoid an obstacle, the emergency sensor system along the perimeter should prevent collision. If both systems fail, an emergency stop signal can be sent via the on board mechanical stop button or the wireless emergency stop button. The large mechanical stop button will be centered in the top panel at the front of the robot for easy access. The wireless stop will be achieved using a radio frequency module with a working range of 50 meters.

(5) BUDGET

Many of the most expensive components required for a successful design were purchased or donated last year. Among them the most expensive are the laser rangefinder (LRF) and the global positioning system (GPS); these two components account for nearly half of the total project cost. In addition, the motors, batteries, camera, motor controller and battery chargers have been purchased. It is important to note that each component used from last year's model was rated against the design requirements as well as several alternative options. For all of the aforementioned components, the fact that no extra cost would be imposed on the

team was a considered only after assuring that the component would meet or exceed the design requirements.

Though many components from last year could be used, others could not. This year the computer will consist of several independently selected components as opposed to a pre-packaged system. This includes a Mini-ITX Motherboard, nVidia8800GT video card, Core 2 Duo 1.66-1.8GHz processor, DC-DC power supply and compatible hard drive and RAM. These components will collectively cost approximately $1000. In addition to the computer, there are multiple components of the chassis that must be purchased. The front tires, wheels and hub attachments along with the rear caster tires and springs will cost approximately $150. Aluminum-6061 tubing and costs associated with the fabrication of the frame, including Aluminum-4043 filler welding rod, will cost approximately $400. Significant cost savings have been achieved for the testing of vibrations through the donation of a DataPAC 1200 and connecting accelerometer. An additional $100 had been allotted for miscellaneous supplies which may be necessary during fabrication and testing of the robot.

Finally, the travel expenses have been estimated for the four-day trip to Michigan for the competition. If the team drives, the trip will be extended to an 8-day trip. Cargo van and trailer rental for 8 days will cost approximately $350 and gas for the nearly 2500 mile round trip will cost approximately $550. Hotel accommodations will cost approximately $450 with a discounted rate of $49/rm/nt (complimenst of Sheraton Hotels). Adding in funds to cover some of the food expense would make the total cost approximately $1800. A second option would be to fly the team to Michigan and ship the robot. This option may be equally expensive and much less time consuming. A full chart of the proposed cost breakdown is shown in the following figure.

Cost Breakdown

Description	Estimated Cost
Laser Rangefinder	$2700 (Purchased)
GPS	$3000 (Purchased)
Motors	$600 (Purchased)
Camera	$300 (Purchased)
Batteries & Chargers	$500 (Purchased)
Motor Controllers & Encoders	$700 (Purchased)
Testing Equipment	$1000 (Donated)
Computer	$1000
Frame Material, Wheels, Casters, Tires	$550
Miscellaneous Supplies	$100
Travel & Accomodations	$1800
TOTAL:	**$12,250 (Remaining Costs=$3450)**

(6) SUMMARY/CONCLUSION

The2008 Louisiana State University Intelligent Ground Vehicle introduced in this report provides a functional design that is expected to successfully attend the next IGVC. From its unique path planning methods to its practical chassis, the design of MikeRobot is a collaboration of two teams: mechanical and electrical. As it was discussed in this design report, the electrical team tackled issues such as obstacle detection, path planning and hardware/software required. The mechanical team successfully addressed the design of a new lightweight chassis, reduced camera and electronic vibrations as well as catered to the needs of the electrical team's requirements. The four-month design process undertaken by the IGVC team can be summarized as follows:

1. A method for detection and mapping of obstacles was successfully devised.
2. A unique path planning algorithm is in progress for implementation.
3. The image processing load was transferred from the CPU to the GPU.
4. The chassis was modeled using FEA and is acceptable for manufacturing.
5. Vibrations and camera movement have been reduced according to the performed analysis.
6. The vehicle dynamics analyzed are acceptable for 5 mph sliding, turning and tipping.
7. The Laser Range Finder will be protected from humidity using innovative heat transfer methods.

Other issues such as future testing plans, manufacturing methods and safety were also discussed in this report. Overall, the IGVC team believes that the design of MikeRobot has been completed and is ready for algorithm coding and manufacturing.

(7) REFERENCES

[1] Association of Unmanned Vehicle Systems International. < http://www.auvsi.org>.

[2] Intelligent Ground Vehicle Competition. < http://www.igvc.org>

[3] IGVC Document: <http://www.igvc.org/rules.html>

[4] van de Weijer, Joost; Schmid, Cordelia; Verbeek, Jakob, "Learning Color Names from Real-World Images," Computer Vision and Pattern Recognition, 2007. CVPR '07. IEEE Conference o*n* , vol., no., pp.1-8, 17-22 June 2007

[5] http://lear.inrialpes.fr/people/vandeweijer/color_names.html

[6] AX3500 Motor Controller User's Manual. pp.126

(8) ORGANIZATION AND ACKNOWLEDGEMNETS

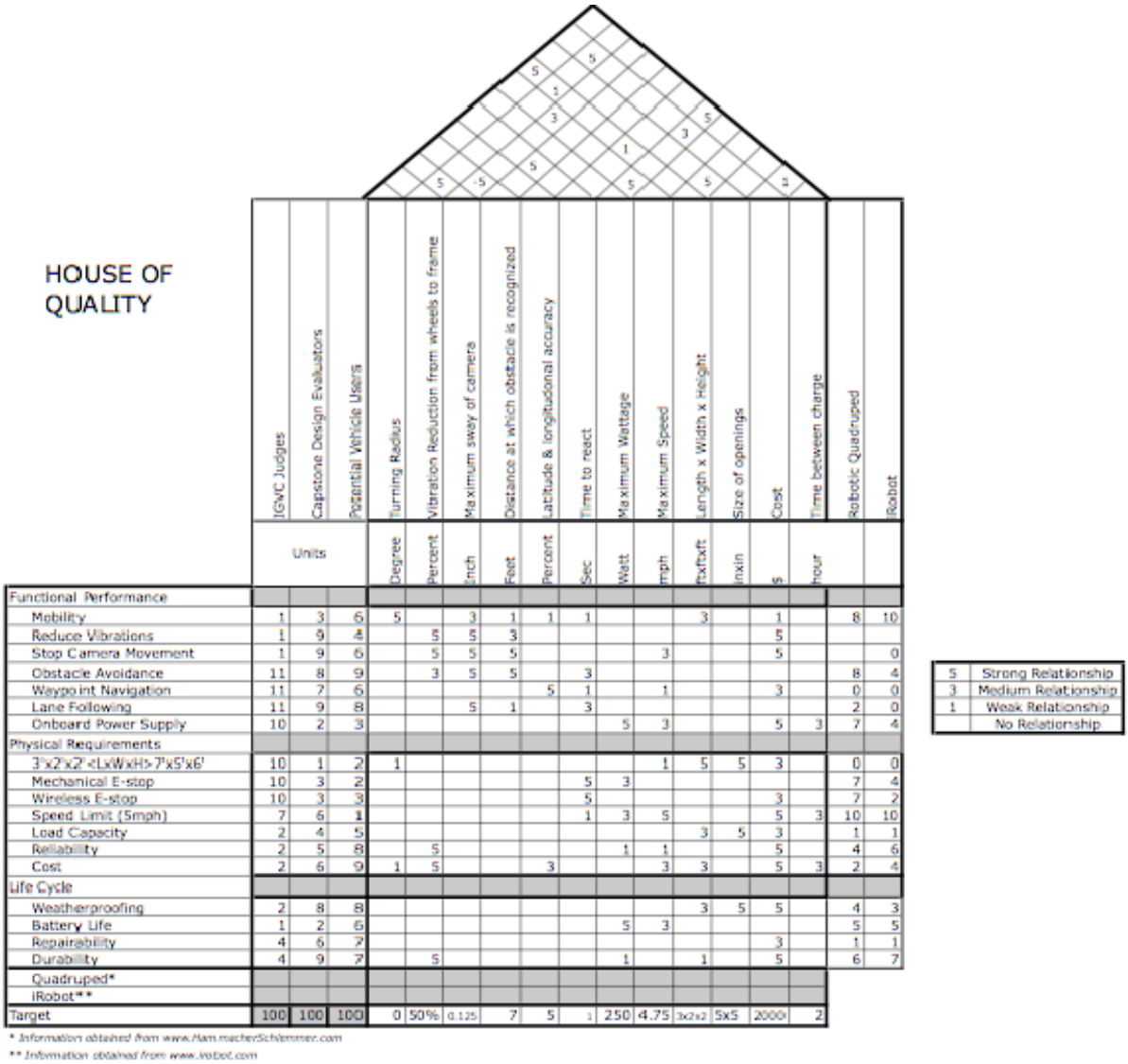
The IGVC team is divided into two groups, a mechanical (ME) group and an electrical (ECE) group. Obstacle detection, line detection, wave point navigation and the evaluation of best routes are some examples of tasks that will be addressed by our ECE group members. Specifically, William Burke will be handling image visualization, and path planning. Alex Ray will help with path planning and also manage GPS navigation and motor controls. Finally, Geoff Donaldson will focus on purchasing and building the hardware and software platforms.

Although the electrical focus varies from the mechanical one, it is important that the vehicle operates as one unit. For this reason, the mechanical design is to be molded around the needs of the electrical design. This includes focus on vibration reduction, maneuverability, power distribution, weight distribution, structural integrity and weatherproofing. Diego Gonzalez will handle chassis and power distribution while David Mustain will focus on maneuverability and vibration reduction.

We would like to take this opportunity to acknowledge our advisors for their help and support throughout the first stage of this year-long design process. The design of the prototype discussed in this report could not have been successfully developed without their guidance. We would also like to thank Shell Chemicals for their assistance with technical equipment. Similarly, we are grateful for the discounts offered by Polaris and Metals Depot when purchasing their parts. Finally, we would like to express our gratitude to the LSU faculty and staff that have provided us assistance.

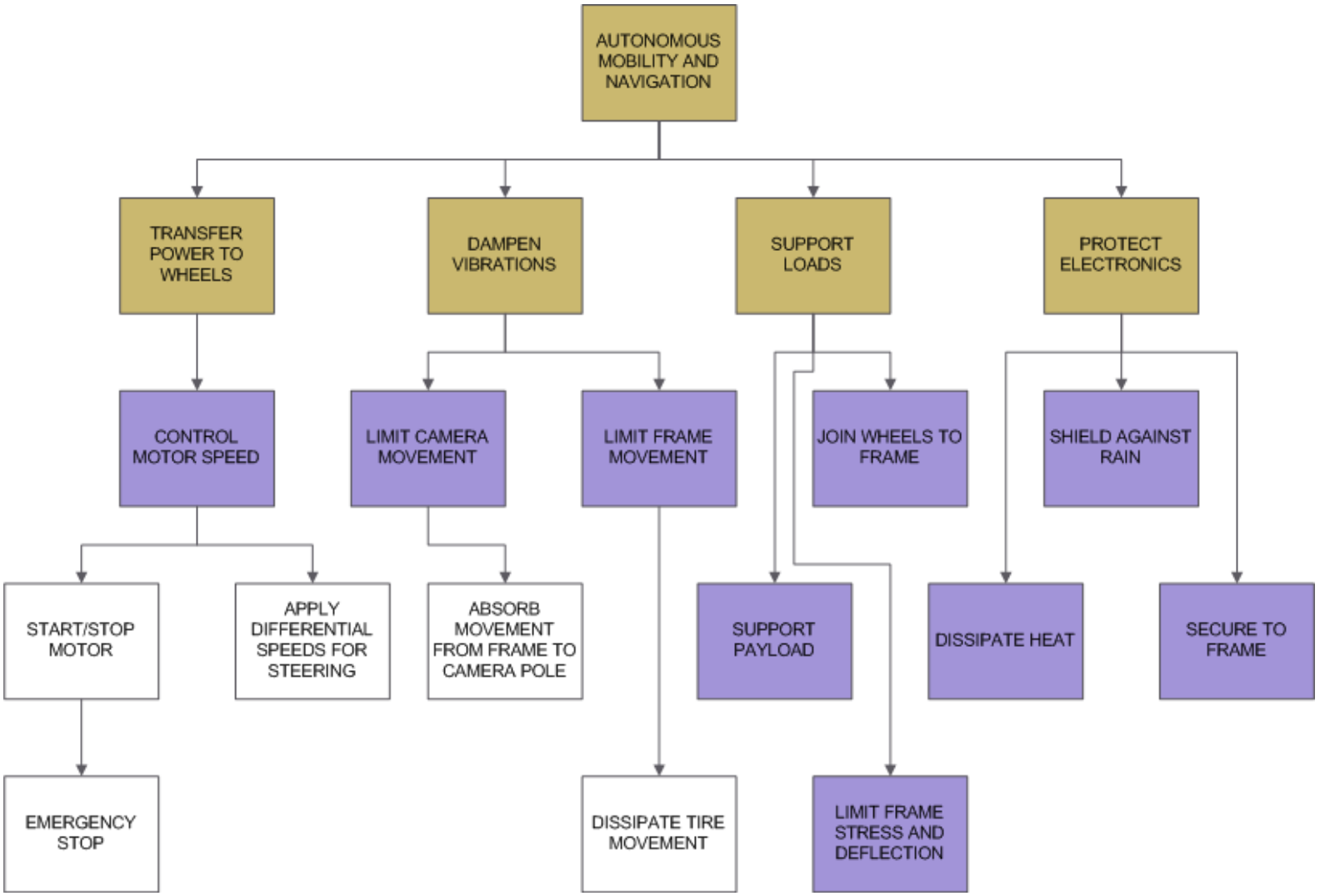
(9) APPENDICES

(A) ME Team House of Quality

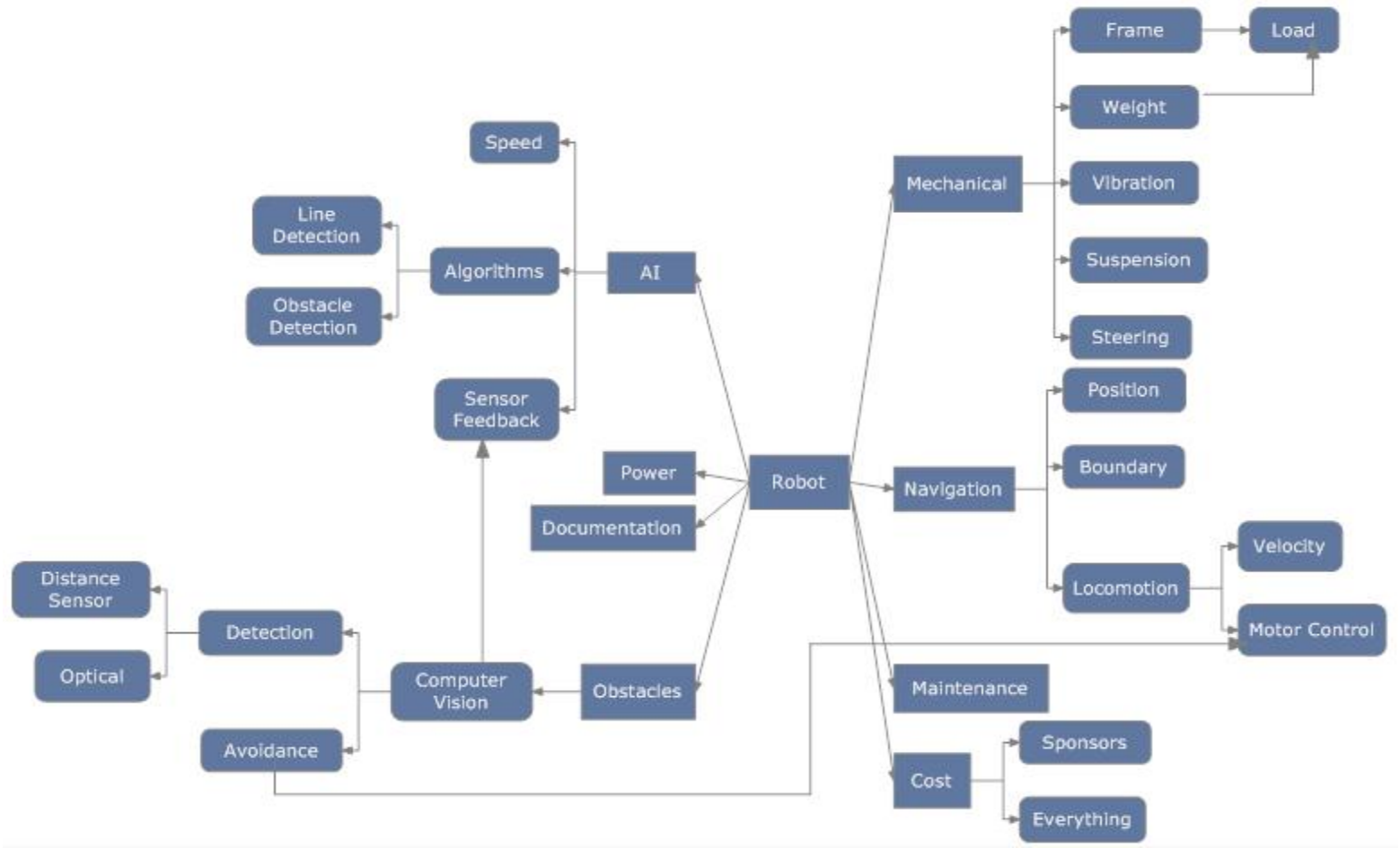


The House of Quality relates the engineering specifications to the performance, physical and life requirements of the vehicle. A total of 100 points were distributed among all measured categories (i.e. Functional Performance, Physical Requirements and Life Cycle) for each of the vehicles customers. The engineering specifications at the top of the house are related to each of the requirements by a point scale of 5 being a strong relationship and a blank space for no relationship. The same scale is followed for relationship among engineering specifications.

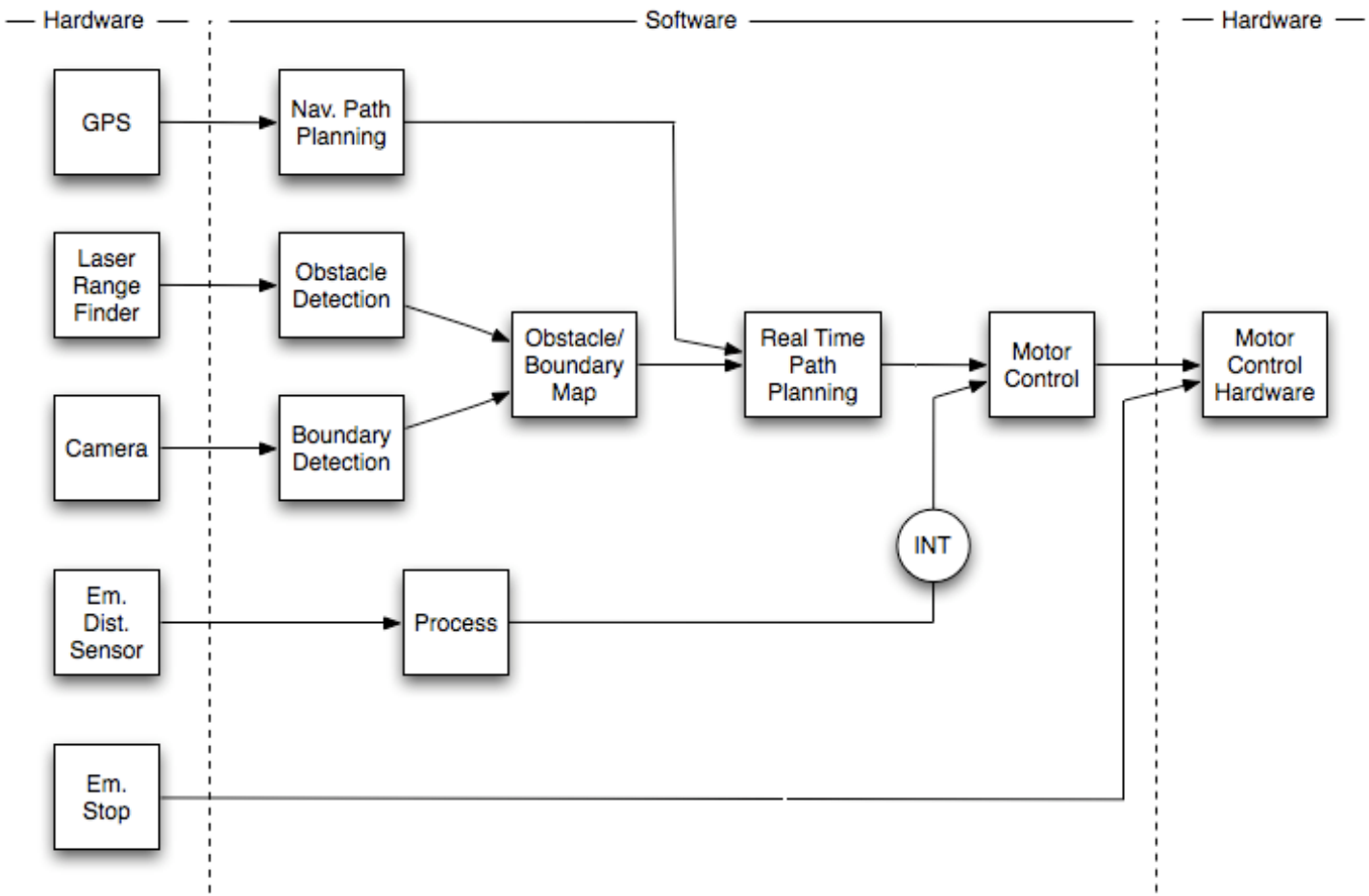
(B) ME Functional Decomposition



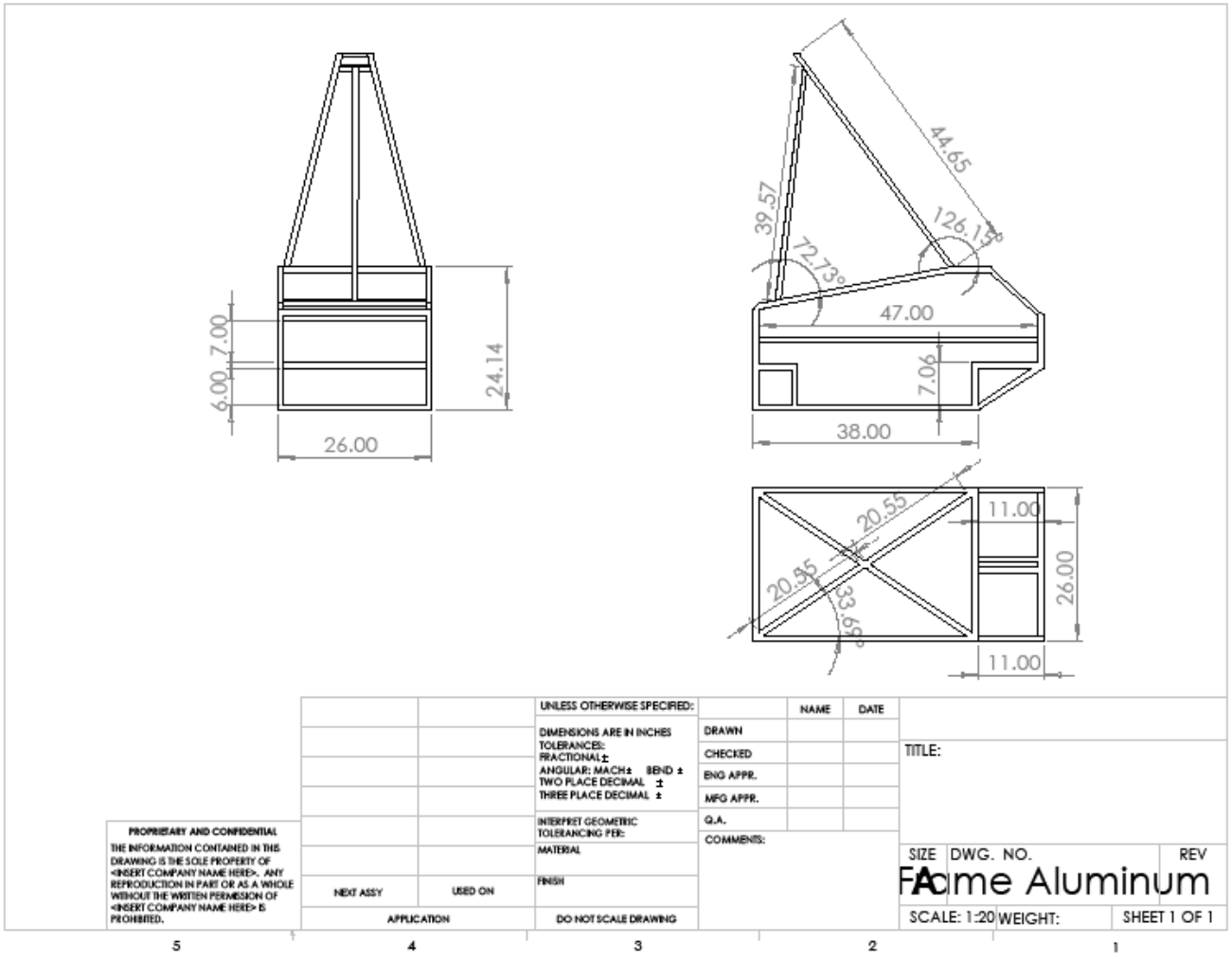
(C) ROBOT MINDMAP



(D) TOP LEVEL SYSTEM DIAGRAM



(E) CHASSIS MANUFACTURING DRAWINGS



(F) POWER REQUIRED CALCULATIONS

Input			Tire radius (in)=	6
Vel. (m/s)=	2.2352		RPM (5 mph)=	140
theta=	0.261799388			
g=	9.81			
f=	0.06			
Mass (kg)	**Power (Watts)**	**Hp**	**Torque**	**Vehicle Weight (lbs)**
20	138.92	0.19	6.99	44.09
30	208.38	0.28	10.48	66.14
40	277.84	0.37	13.98	88.18
50	347.30	0.47	17.47	110.23
60	416.76	0.56	20.97	132.28
70	486.22	0.65	24.46	154.32
80	555.68	0.75	27.95	176.37
90	625.14	0.84	31.45	198.42
100	694.60	0.93	34.94	220.46
110	764.06	1.02	38.44	242.51
120	833.52	1.12	41.93	264.55

$$\boldsymbol{Power = mgV(sin(\theta) + f_k cos(\theta))}$$

$$\boldsymbol{Torque = \frac{HP \times 5252}{RPM}}$$

(G) MOTOR SPECIFICATIONS

Torque in lbs	Amps	RPM (20:1)	HP
30	8.6	238	0.11
60	12.5	230	0.22
90	16.2	225	0.32
120	20	218	0.41
150	23.5	211	0.52
180	27.5	206	0.62
210	31.6	200	0.71
240	35.1	194	0.81
270	39.2	187	0.89
300	43.1	181	0.95
825	110	STALL	

(H) MOTOR DRAWINGS