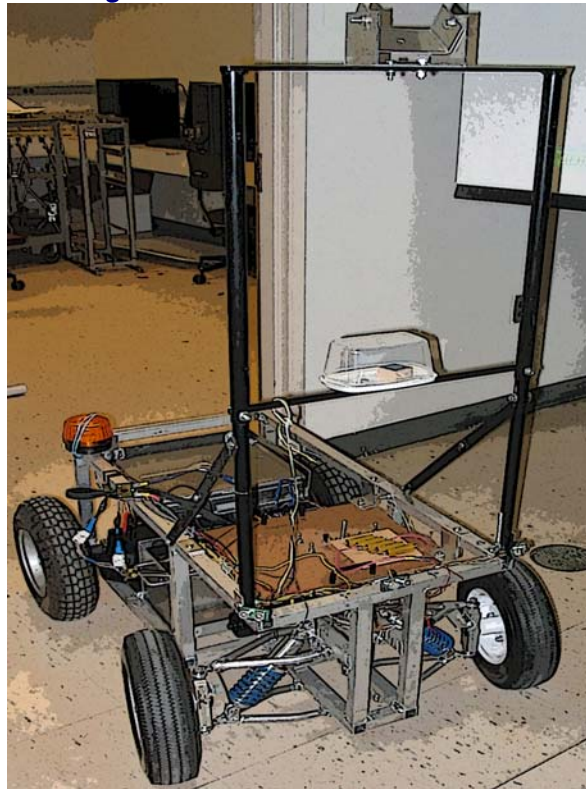




The Autonomous Vehicle Team from TCNJ Presents:

NJAV

New Jersey Autonomous Vehicle



Team Members

Mark Buechel, Terence Nish, Nicholas Vacirca

Required Faculty Advisor Statement:

I hereby certify that NJAV is a new vehicle designed and built completely from scratch by the first team from The College of New Jersey. The team consists of seven undergraduate students with mechanical, electrical, and computer engineering backgrounds. Each member received four credit hours in senior project credit for their work on this project.

Dr. Orlando Hernandez, Department of Engineering, The College of New Jersey
Faculty Advisor

Introduction

The team members of the New Jersey Autonomous Vehicle Team are excited to present the product of their year's worth of hard work. This vehicle, designed and built from the ground up, was constructed and tested for the sole purpose of competing in the 14th annual Intelligent Ground Vehicle Competition. The team spent several hours researching options for their completely innovative design of the vehicle, while also keeping in mind their financial limitations since they are the second team from The College of New Jersey to enter the IGVC.

Design Process

The team members of NJAV formed our project team in September of 2005. At this meeting, team jobs were assigned, tasks were agreed upon, and we discussed the progress and work of last year's team and previous teams that had competed in the IGVC. The team jobs were assigned as follows:

Mark Buechel	Senior Computer Engineering	Hardware Systems
Terence Nish	Senior Mechanical Engineering	Mechanical Systems
Nicholas Vacirca	Senior Electrical Engineering	Software Systems

It was at this time that our team decided upon its goals and what exactly we wanted to accomplish as a novice team entering into the IGVC. Since last year's team qualified for the competition but was unable to compete due to technical difficulty, the team's goal for this year was to qualify for the competition and compete for a top rank in the obstacle course, the navigation challenge and the design competition.

MECHANICAL SYSTEM

Frame Design

At the root of all objects used in transportation are its frames. The frame is the infrastructure to which all other systems on the object are joined and integrated. The NJAV is no different. Due to these major responsibilities of the frame, it was the first priority to design the frame to be fully capable of performing its duties.

The frame was constructed from 6061-T6 1-inch square aluminum tubing with a wall thickness of 0.0625 inches. This material was chosen due to its high strength to weight ratio and its ability to be welded, machined and formed with quality results. A CAD drawing of the frame is shown in Figure 1. The dimensions of the frame are 35.5 inches long, 18 inches wide and 14 inches tall. This design is optimal since it maximizes the internal volume while minimizing wheelbase and track width. The increased volume allowed for all components to be modularized and placed inside the protective confines of the frame while reducing the overall length and width of the vehicle. The modules made clear delineations of parts and components to ease assembly, disassembly and debugging while the reduction in width and length of the vehicle aided an increase in maneuverability.

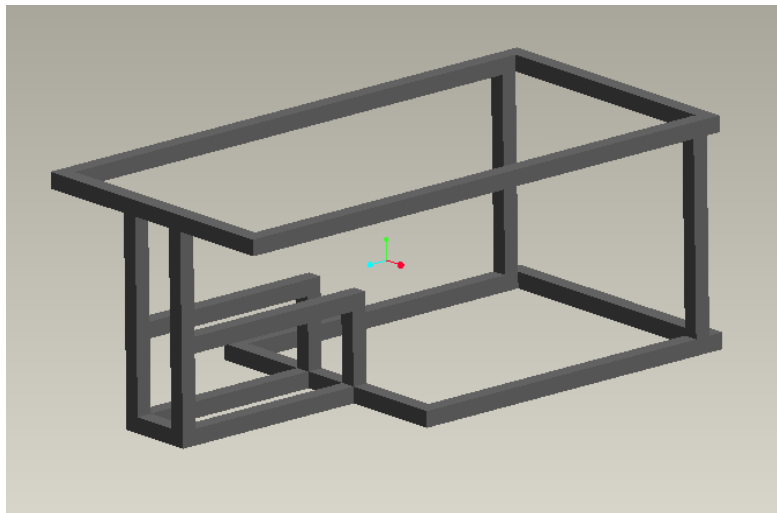


Figure 1 2006 NJAV frame design

Chassis Design

The design of the chassis includes the frame design along with the drive train, the wheels and tires, the suspension system and the steering system.

Drive Train

Through the generous donation from Mr. Alex Michalchuk, technician at TCNJ, the team acquired a motor suitable for NJAV. This motor features a built-in gear reduction unit that provides a final shaft speed of 175 RPM, a true differential between the left and right axles, and a housing that is nearly the same width as our desired vehicle frame. A picture of NJAV's drive train can be seen in Figure 2.

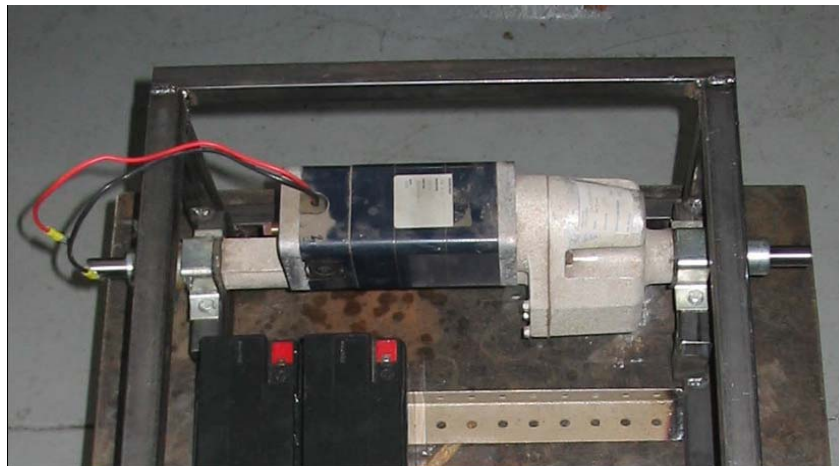


Figure 2 The drive train used in NJAV.

Using this motor, NJAV is able to traverse the ramp inclines it will encounter during the competition by using its maximum power output. Through several calculations, using our estimated weight of approximately 100 pounds, we discovered our motor needed to have 0.2 horsepower or 150 watts at 2000 RPM (or greater power output) in order to power NJAV. This drive train successfully accomplishes this task.

Suspension System

The suspension system was designed for two purposes. The first was to ensure that all four tires remain in contact with the ground. Due to the type of differential in the drive motor, if some wheels lost contact with the ground when the vehicle encountered uneven terrain the vehicle could become disabled or interpret false information about its oncoming course.

The second purpose of the suspension system is to offer some vibration control and shock absorption that is experienced by sensitive electronic components onboard such as the camera, the laptop and the GPS navigation system. Isolating these components from vibrations experienced by the chassis improve there ability to function better during the competition.

The design for the suspension system is a parallel a-arm independent front suspension. It is shown in Figure 3.

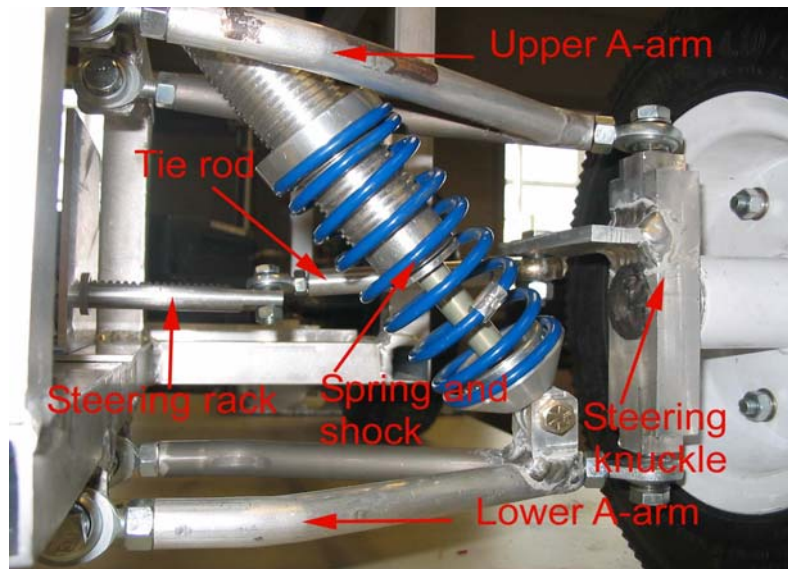


Figure 3 Suspension system

This type of system was chosen because it is simple, easy to manufacture and provided adequate absorption. It has a travel of 1.5 inches. This was calculated using the weight of the vehicle and the assumption that the maximum deflection would occur if the vehicle fell into one of the 2 inch deep potholes located on the obstacle courses.

Steering System

A Rack-and-Pinion steering system was implemented in conjunction with a digital rotary encoder. This is a robust steering system that takes up a small amount of space. It makes use of a small pinion gear to allow uniform and stable translation of the rack and in combination with the encoder allows the vehicle to always know in what direction the tires are pointed. The steering system is shown in Figure 4.

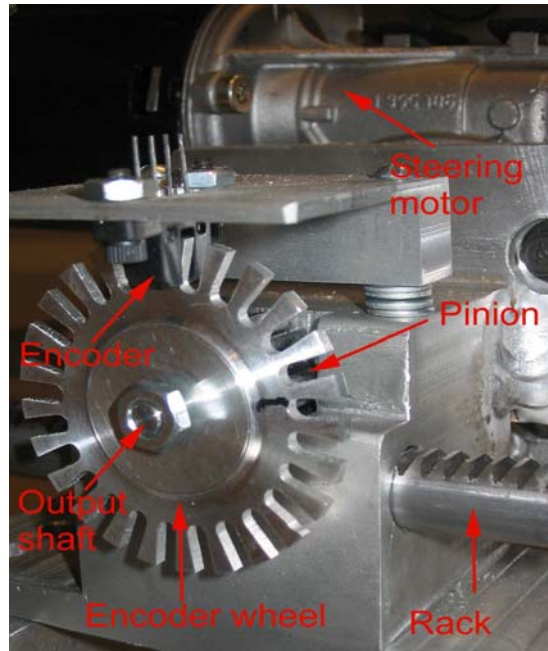


Figure 4 2006 NJAV steering system

The steering geometry design for the vehicle is Ackermann geometry. It allows both wheels to turn about a radii that share a common center. Ackermann geometry is most effective in low speed applications. Since competition rules state the car cannot exceed a speed of 5 mph, the vehicle will be operating in a low speed environment.

HARDWARE SYSTEM

Digital I/O

In order for the vehicle's Laptop to be able to communicate with all of the peripheral devices that make up NJAV's various systems, a digital I/O board was used. The particular board used was the National Instruments DAQCard-DIO-24, which is a PCMIA card that plugs directly into the laptop and has three separate 8-bit ports that are I/O configurable. The card, shielded cable, and connector block are shown below in Figure 5.



Figure 5 PCMCIA card, shielded cable, and connector block

Obstacle Detection System

The obstacle detection system implemented on NJAV consists of two different components, a digital camera and ultrasonic sensors. The digital camera is discussed in more detail in the image processing section of this report. The sensors used are the Devantech SRF04 Ultrasonic Ranger sensors that have a range of 3 inches to 10 feet. Three of these sensors are mounted on the front end of the vehicle in the configuration shown below in Figure 6.

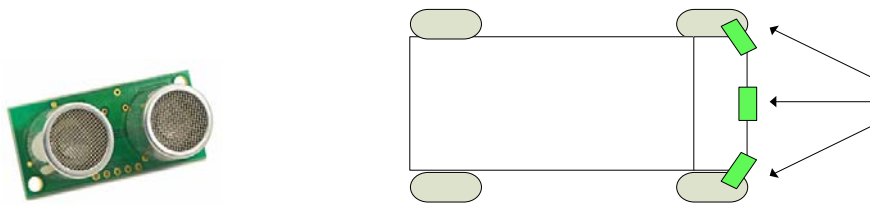


Figure 6 Sensor arrangement on front of vehicle

The sensors operate by emitting an ultrasonic pulse and then software is used to determine how long it takes for the return of that pulse to reach the sensors. Based on the timing the software is able to determine the distance to any objects in the sensor's field of view.

Steering System

The steering system implemented for NJAV consists of a 12V DC motor that is driven by means of an H-bridge circuit. The H-bridge circuit shown below in Figure 7 is controlled by connecting outputs from the I/O board to BJT switches that then allows current to flow through a series of relays and then be sent through the motor to turn it in the proper direction.

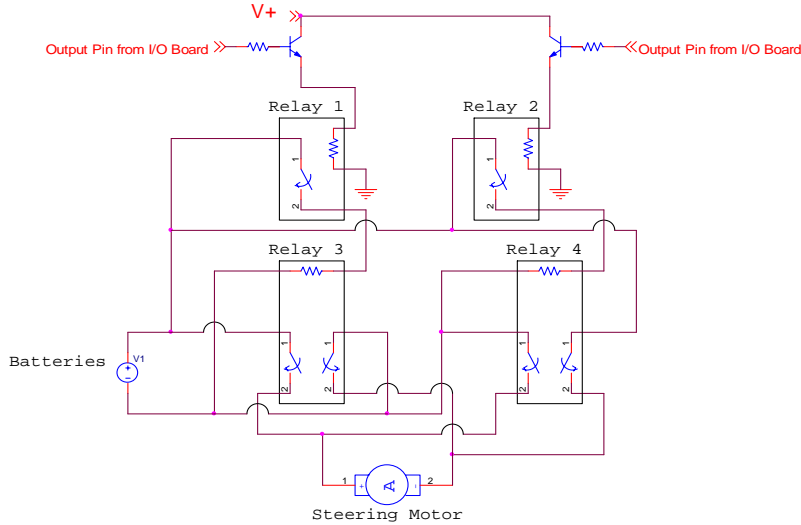


Figure 7 H-bridge steering circuit

In order to ensure that the computer can control the steering a digital encoder is used to determine how much the wheels have turned. The encoder is made up of a phototransistor and a notched photo-interrupter disk. The disk is attached directly to the output shaft of the steering motor and the encoder is positioned in such a way that as the motor turns the disk, the notches pass through the phototransistor and break the beam that produces a square wave similar to the one shown below in Figure 8. The computer counts the number of pulses and is able to determine the position of the front wheels.

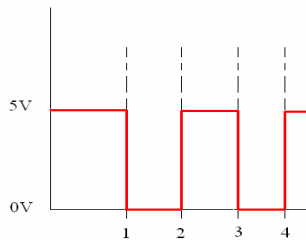


Figure 8 Square wave output read by encoder

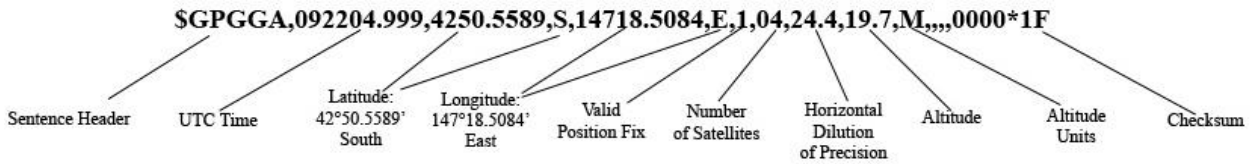
Navigation System

The navigation system requires the use of a GPS locator as well as compass in order to determine the direction the vehicle is facing. The Fastrax uPatch100 GPS locator was used as it provides the NMEA 083 standard sentences shown below in Table 1 as available outputs.

Table 1

Heading	Information
GPGGA	Global positioning system data fix
GPGSA	Active satellites being used
GPGSV	Satellites in view
GPVTG	Course over ground
GPZDA	UTC and local date/time

Of the uPatch100’s available outputs, the GPGGA NMEA 083 sentence was selected, as this was the only sentence that provided the latitude and longitude of the unit. The output produced by the unit is shown below, where the standard GPGGA sentence is broken down into its various fields; the computer extracts the latitude and longitude components of this sentence to determine the vehicle’s current position.



For the digital compass a Dinsmore 1490 was selected for its low cost, small size, and ease of integration. The Dinsmore 1490 also came with a simple test circuit schematic that could be used to determine if the compass was working properly before being integrated into a system. The circuit diagram is shown below in Figure 9.

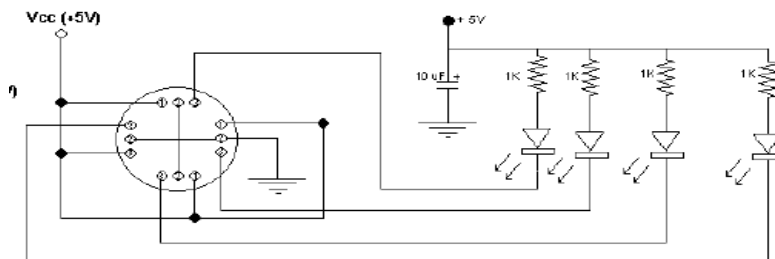


Figure 9 Circuit diagram of Dinsmore 1490 test circuit

This circuit was constructed and used to determine that the compass was in fact working properly based on the LED lighting patterns which matched the specifications of the

circuit as is shown in Figure 10, the circuit was then adapted to interface with the laptop as is shown below in Figure 11.

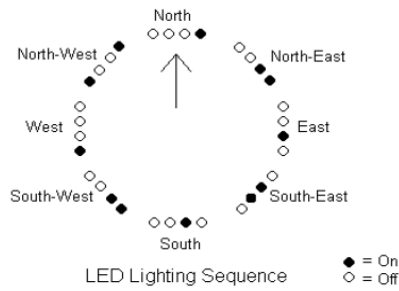


Figure 10 LED lighting sequence for test circuit

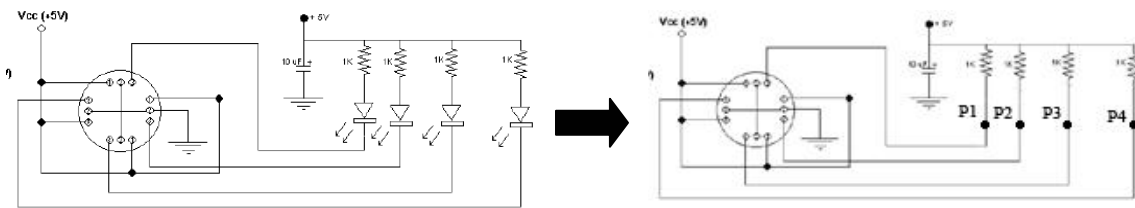


Figure 11 Compass test circuit adapted in order to connect to the laptop

Now that the hardware for the GPS navigation system had been decided up the algorithm, which would be used for the GPS navigation system could be developed, and the code for it written using MATLAB. A flow chart of the algorithm developed for this system is shown in Figure 12.

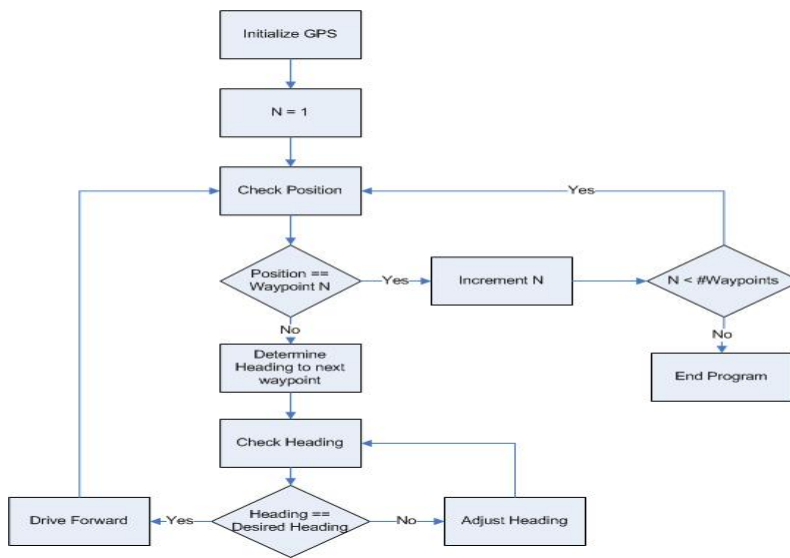


Figure 12 Navigation algorithm

SOFTWARE SYSTEM

Ultrasonic Sensor System

The development of an intelligent ultrasonic sensor system that would be controlled solely by the team laptop is crucial to NJAV's physical obstacle detection. A field consisting of ten discrete sensor output levels was established, and the real-world distance corresponding to each quantized range was found. This creates a field of vision to allow the sensors to "see" obstacles in its immediate path. Depending on the location of obstacles, an intelligent decision can be made to avoid contact with the obstacle(s) through various degrees of steering, or whether the vehicle should move in reverse and make another attempt.

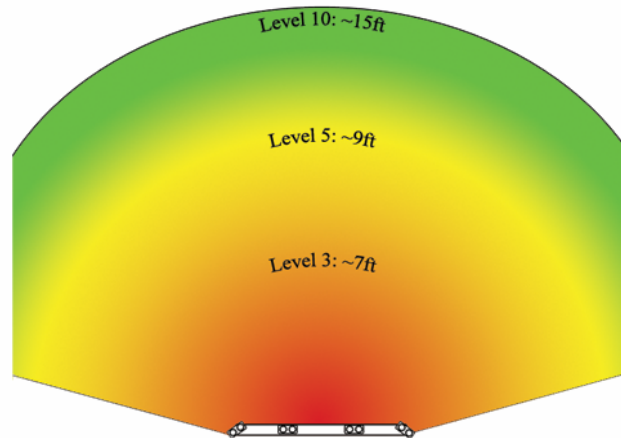


Figure 13 Sensor field of view

Image Capture Device

The team decided that the current webcam-based image capture device was inadequate for the task at hand. Several options were explored, such as a higher quality webcam, a machine vision type camera, a digital camcorder, and a digital still camera. Each option has both merits and downfalls, which will be expanded upon. Although the webcam did provide an effortless means of image acquisition with built-in support from our Matlab platform, its maximum image resolution and quality were severely lacking. The option of replacing the webcam with a higher quality camera of the same type was explored; however webcams are usually aimed at indoor use and low-resolution video

conferencing/chatting. More sophisticated webcams employ auto-white balancing and some enhancements through proprietary software, but these are inaccessible to our Matlab interface.

Machine vision and digital video cameras, were both discredited for similar reasons. Machine vision cameras use top-of-the-line CCD sensors and supply excellent quality video streams. Digital video cameras, aimed at a consumer market, also supply high quality video streams. Both types of cameras are available over IEEE1394 Firewire interfaces, and are easily interfaced into our software platform. The cost factor, however, for both of these camera types severely hindered their feasibility. Another drawback to these cameras is their video stream capture. These cameras, as well as webcam devices, are intended to capture a video stream, supplying up to thirty frames per second. For our autonomous vehicle, there are no conceivable circumstances that would demand an image to be captured faster than once every one to two seconds. For a much lower cost, a higher quality and resolution image can be delivered well within this demand by a digital still camera.

Digital still cameras, widely available from myriad suppliers, represent an excellent balance of features, image quality, image resolution, and cost. This balance gives it an edge as seen in the decision matrix below.

Image Capture Device Decision Matrix								
	Hardware Features	Resolution	Image Quality	Ease of Interfacing	Cost	Mounting	Speed (fps)	Total
Webcam	1	2	1	5	5	2	5	21
Machine Vision Camera	1	4	5	5	1	5	5	26
Digital Video (camcorder)	3	3	3	5	3	4	5	26
Digital Still Camera	5	5	5	2	4	5	3	29

The Canon Powershot A60 digital camera was available to the team free of cost and with immediate turnaround, and is capable of delivering maximum image resolution of 1600x1200 pixels at a rate of 2.6 frames per second. In addition to the gain in resolution over the webcam's 640x480 pixel or 800x600 pixel images, the contrast and color

representation from the A60 is unparalleled when compared to images from the previous webcam device (see Figure 14 below).



Figure 14 Captured images from the Logitech webcam (left) and Canon Powershot A60 (right)

The A60 also contains hardware features that will allow it to consistently deliver high quality images across varied course conditions, such as Auto-White-Balance and Auto-Focus. The A60, however, is not without drawbacks. The main negative of the device is that it has no simple way of integrating into the Matlab software platform. Where many webcams/video devices have built in support from Matlab because they are constantly running a video feed, digital still cameras have no such support. With the help of the Canon digital camera software development kit (SDK), an interface was written in Visual C++ to control the camera through a supplied Windows driver. This interface is capable of complete control of all of the camera's functions through a USB connection, and also physically locks the control buttons on the device to avoid any accidental interference. The image is transferred directly from the camera's memory to the computer's hard drive, bypassing the memory card on the camera completely and in turn making the process as fast as possible. This image is saved in the .jpg image format and imported into Matlab for processing. One of the strongest points of the camera's interface is its ability to communicate with nearly all Canon digital still cameras in production (some of the very high-end digital SLR cameras have their own SDK and drivers), allowing future teams flexibility to upgrade and take advantage of developing technologies.

Image Database

Three test courses were created under different conditions, with a full representative image set captured at each location. As seen in the breakdown below, the image database covers different weather and lighting conditions at each of the three courses, which themselves represent different types and conditions that may be encountered.

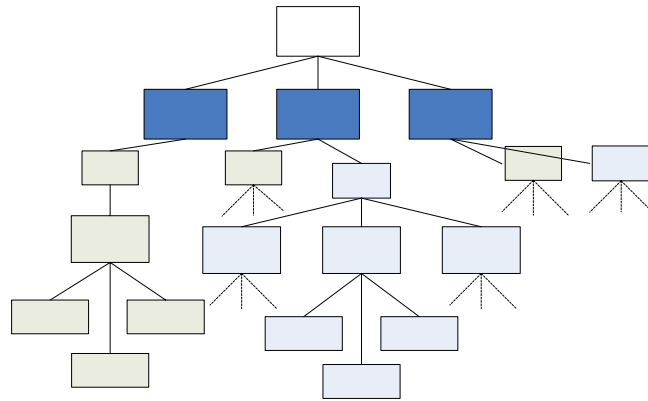


Figure 15 *Image Database breakdown*

Image Processing

The theory behind the image processing and line detection algorithm developed is quite simple. Examining the properties of the color white, its RGB spectrum is unique in that it contains extremely high values of all three primary colors, stemming from the fact that white is the resultant sum of all colors of light. The first step in finding lines in an image is then to simply find the pixels that have values above a certain threshold in all three of the primary color matrices. The result of this first pass can be seen below in Figure 16.

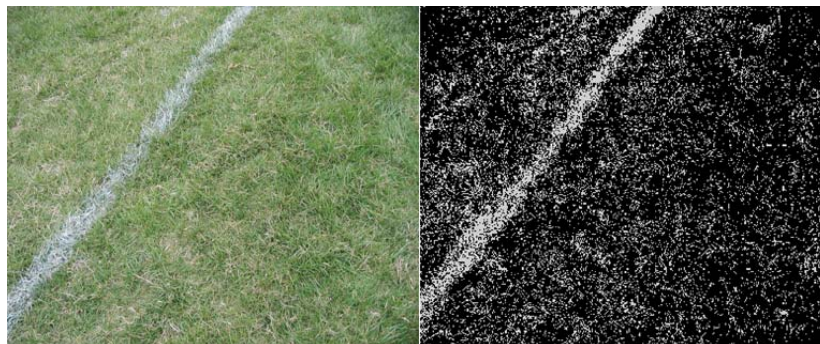


Figure 16 *A captured image (left) and the first pass of pixels viewed as white (right)*

As can be seen, there is a dense concentration of white pixels where the line should be detected; however there is also a great deal of noise. To remove these noise pixels a simple median filter is used. After filtering the image, the noise pixels left are in concentrated areas outside the main line. Matlab’s edge detection function, using the Hough transform, is applied to the image. This yields the edge detection image below and a matrix containing Cartesian coordinates at the endpoints of edges that are detected as lines. Line segments will be examined to see if they pose any threat to the vehicle in a manner similar to the threat levels generated in the ultrasonic sensor system. Slightly different than the sensor system, though, the slope of the line and the proximity of its X-Y intercepts to the vehicle will be used to determine the vehicle’s reaction. Figure 17 shows both the detected edges and the final resultant detected line. Finally, the results of testing the developed algorithm on the compiled image database are shown in Figure 18.

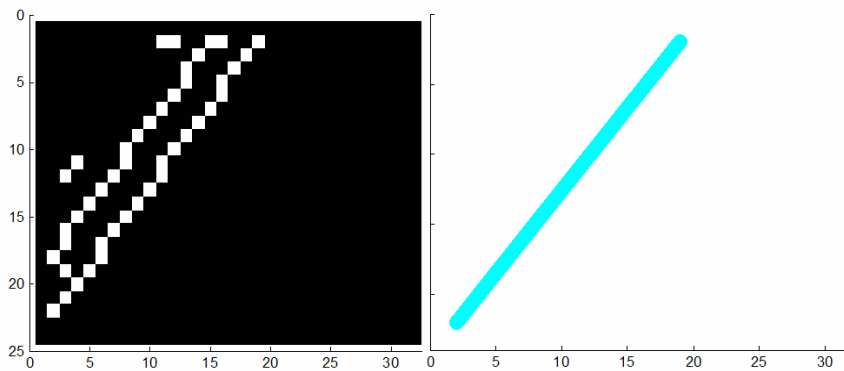


Figure 17 *Detected edges (left) and the resulting detected line (right)*

Image Database Testing Summary

	#Images	Success Rate
WRF Marlton	21	90.48%
Trackside TCNJ	237	72.15%
TCNJ THS	221	90.50%
Other	95	70.53%
Overall	574	79.62%

Figure 18 *Summary of image database testing of the new image-processing algorithm*