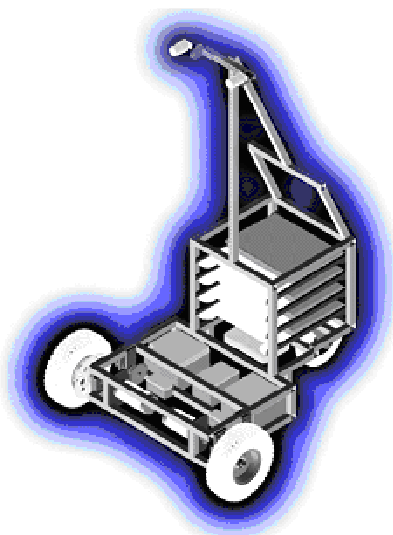
UNIVERSITY OF ILLINOIS AT CHICAGO

"Achilles Autonomous"



Engineering Design Team

Mohammad Ali	BIOE	Sophmore	Leeder Hua	ME	Senior	Rafal Radulski	CS	Junior
Dan Belarmino	EE	Senior	Steve Kearney	ME	Senior	Mihai Rosu	ME	Senior
Tom Burdick	EE	Senior	Jennifer Kinahan	CS	Freshman	Rayna Saito	BIOE	Senior
Matt Byrne	CS	Freshman	Nick Kutella	EE	Junior	Keith Weber	CS	Freshman
Dan Egerton	EE	Junior	Anup Kotadia	EE	Senior	John Wrightsell	EE	Junior
Dalia Garcia	CE	Junior	Nick Novak	ME/EE	Senior			

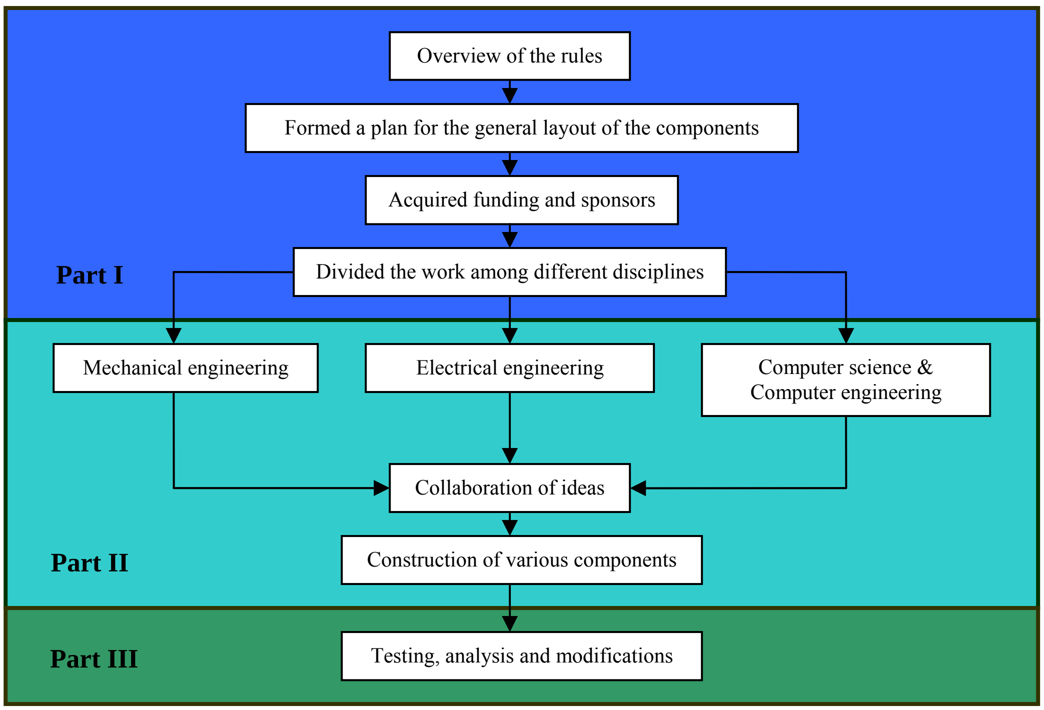
The engineering design in this vehicle by the current student team has been significant and equivalent to what might be awarded in a senior design course.
Faculty Advisor Certification Signature: ___________________________

1.0 Introduction

Achilles is a fully autonomous robot that was built by the Engineering Design Team at the University of Illinois Chicago. An autonomous robot is a complex system of software, electrical, and mechanical engineering. To build Achilles many engineering disciplines collaborated together on this robot. The design process, predicted performance, mechanical, electrical, and software components of Achilles are detailed in this report.

2.0 Design Process

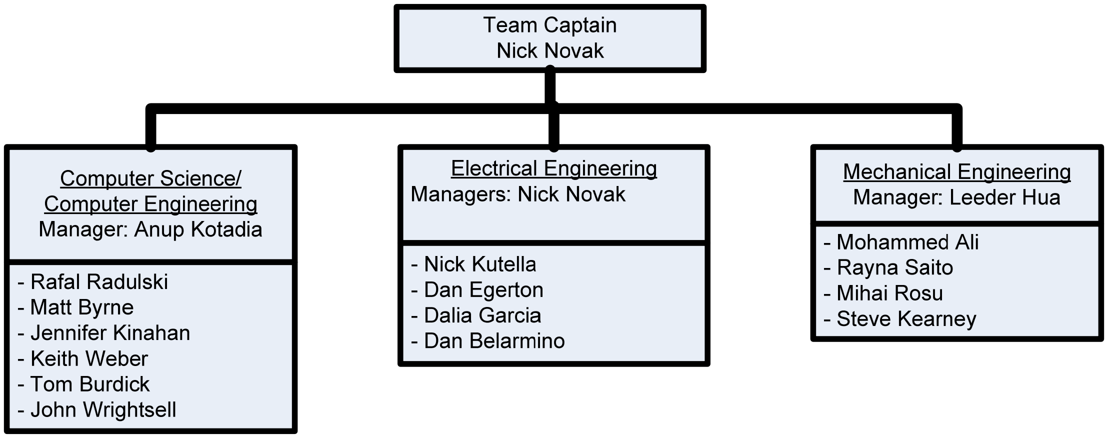
This diagram details the design path followed to construct Achilles.



2.1 Design Process Part

As a team the rules of IGVC were reviewed and discussed. A timeline of approximately 9 months was given to complete this task. The task consisted of everything need to build an autonomous robot, all of the main hardware and software components. An approximate budget of $6000 dollars was thought to be needed to successfully obtain the primary goal. Money was then gathered from within the university and without, through meetings with top company executives to public relations people.

2.2 Design Process Part II



The figure above clearly outlines how the work was managed. The grouping was done into three main groups which correspond to the mechanical, electrical, and software components of Achilles. Frequent meetings of all group members were had to share ideas and progress with other engineering disciplines. These meetings helped to make sure that Achilles would work and there were no major design flaws.

During idea collaboration and construction, particular emphasis was put on the safety, durability and reliability of Achilles. Safety was always an issue, to make sure there were no fires from electrical wiring and so that Achilles could be shut down quickly if acting erratically. Durability was another key concern. A long life span was a priority so as to provide a working platform for the generations of students to come. Good durability is also vital to having a good quality product and key for the practicality of Achilles. Focus was also made to ensure that Achilles could reliably navigate an obstacle course more than once. Most real world scenarios never have the same exact course twice; therefore Achilles must be reliable by adjusting to the new conditions.

2.3 Design Process Part III

Testing and analysis of the new design is the longest and most important. The goal was to assemble a working robot no later than a month before the competition to allow for testing. Priority was given to some systems so that particular components could be tested before this deadline. Currently Achilles is in this stage of the design process.

3.0 Predicted Analysis of Performance

Chart below summarizes the predicted analysis of Achilles.

Analysis Section	Predicted Value	Method of Determination
Mechanical		
Speed	top speed 5mph	Used none variables from motors and gear ratios
Ramp Climbing Ability	53°	Set up a 12 to 1 gear ratio
Electrical and Software		
Reaction Times	.1 sec	This was determined through the Controls software component, which is the final output and limiting factor
Battery Life	Max operating time 1 hour	Determined through analysis of battery amp hour rating
Distance at Which Obstacles are Detected	Approximatly 30ft	Predicted from initial trials of Disparity Map component
Complex Obstacle Navigation		
Switch Backs / Center Islands	Can see beyond obstacle using vision	Stereo vision and path planning should make this possible
Traps / Dead Ends	Path planning might deal with these obstacles	Unkown, results testing will clarify
Potholes	Hough Transform, should navigate fine	Works on paper testing will give better results
Navigation		
Waypoint Arrival Accuracy	30cm	Accuracy of the GPS unit

4.0 Mechanical Design

The main goals for designing the vehicle were simplicity, safety, durability, and reliability while being able to achieve its objective of enclosing and protecting all of the systems necessary for autonomous navigation. Simplicity in design allows for undemanding manufacturing processes, reduction in costs, and minimization of potential failures. Accessibility is crucial for trouble free repair and maintenance. Durability is necessary to represent a real world application. Lastly, reliability of all components was worked into the design.

4.1 Mechanical Innovation

There are three specific innovative mechanical designs to Achilles. New drawers that allow access to laptops while still in Achilles were designed. A GPS mount which allows exceptional ease of use and free sky visiblity was designed. Finally, an adjustable sonar mount was designed.

The four laptop computers are stored in the rear compartment. Three key constraints of drawer design are: stability, motion range, and vibration. The constraints were optimized using high capacity telescopic ball bearing slides. The stability from these drawers gives a sound platform for the laptops. The large range of motion allows users to use their laptops while Achilles is still in the field. Vibration pads were installed to protect the laptops.

The sonar mount is located at the very front of Achilles. It was manufactured from aluminum for light weight and durability. It is mounted to Achilles by an adjustable pin joint. The pin joint allows for rotation of the sonar sensors so that when Achilles approaches a ramp the sensors don't trigger on the ramp. Adjustability allows for adjustment to the optimum angle. This optimum angle is a combination of ramp angle and object height.

For the GPS to receive a good signal it is mounted at the top of Achilles were nothing else obstructs its view of the sky. There is a large magnet on the bottom of the GPS unit for mounting purposes. The magnet is strong enough to keep the GPS mounted through rough driving conditions, and it is weak enough to allow for easy removal. A steel plate was attached to the top of Achilles to provide a flush surface for the GPS magnet.

4.2 Chassis

The vehicles frame is constructed out of alloy 4130 military grade 1" by 1" steel tubing. Due to our experience with steel tubing, it became the material of choice. The relative ease of creating any length of tubing as well as creating angled cuts makes manufacturing quick and simple. A welded frame can be cut and re-welded, allowing for quick solutions to unanticipated design faults. The vehicles frame is a demonstration of a design that can be very rugged, durable, and strong, while also keeping it lightweight.

4.3 Vehicle Interior and Compartments

With all of the considerations for what was going inside, it was important to make sure that there was adequate space for accessibility. A majority of the components that comprise the vehicle are contained inside the frame and it becomes important to be able to conveniently access them. The frame is composed of two steel tubing boxes that are connected along an edge. A box frame combines good stability and strength with a simple large volume. Since most components themselves are rectangular or approximately rectangular in shape, it becomes natural to create a housing which is of a similar shape. This also permits rearrangements of the internal components if necessary.

The frame is partitioned into two compartments, the lower front, and the upper rear. The lower front compartment was designed specifically to house the batteries, motors, motor controllers, and the drive train. All of the heaviest components are nearest to the ground to lower the center of gravity and to keep most of the weight on the two front axels, relieving the caster wheel of excess weight. The upper rear compartment houses all of the laptops. In between the two compartments is an intermediate plenum which joins them. It is mainly used for routing cables to reduce clutter and for the placement of the hub that networks the laptops.

4.4 Vehicle Exterior

With all of this open space, there comes the problem of exposure to the elements. An important objective is to allow the vehicle to be able to operate while it is raining. With that in mind, the vehicle needs to be water tight, but at same time be easily accessible. The frame is covered in sheets of polycarbonate and aluminum. Sheets of aluminum, ⅛'' thick, are used to cover the bottom of the vehicle. It provides enough material to sound protection while allowing the vehicle to maintain its lightweight performance. The top is covered in an opaque polycarbonate, ¼" thick, to prevent direct exposure to the sun's rays causing unnecessary heating. Transparent polycarbonate, of the same thickness, is used to cover all of the other sides. Similarly to the aluminum, it serves as protection without adding too much weight. While the transparent look of the vehicle adds to its aesthetics, it also serves the purpose of allowing us to see inside to verify the condition of all parts and wiring. Most of the polycarbonate is attached using latches for quick removal. The other covers are attached directly to frame using weld nuts and machine screws, directly mounting them to frame. Every place there is polycarbonate directly mating with the frame, there are rubber gaskets in between.

4.5 Enclosure Ventilation

One of the major concerns is keeping the upper compartment cool. Heat generated from the laptops must be exhausted, but the vehicle also must be kept water tight. Cool air convection by fans is the method used to cool the laptops. Each laptop is arranged so that the heat outlets all face the same direction. In between each laptop there is empty space through which cool air flows by fans placed at the ends. The fan inlet and outlets are covered by screens, to prevent water from entering.

4.6 Drive Train

The vehicle features a sprocket and chain drive chain. The chain driven system allows freedom in placement of the motors and wheels. On each side of the lower compartment is the drive train for that side. Each wheel has its own motor enabling the wheels to turn in opposite directions. This tank drive provides a much needed zero turning radius for ease of obstacle avoidance. Choosing the tank drive therefore not only makes the navigation of the robot less computationally intensive but also reduces the mechanical complexity of the system as opposed to using steering.

5.0 Electrical Design

Achilles' wiring is concise and simple. The electrical components were located at specific locale for simplification and efficient use of space. All of the circuit boards and sensors are modular so if a malfunction occurs, easy isolation and repair can be accomplished.

The batteries are positioned roughly in the center of the robot for a center of gravity close to the ground. It also minimizes average distance to all powered components. All power cables and signal wires are run through Panduit wiring duct to free up Achilles from cluttered wires. The battery charger and wireless E-stop circuit are located in the back of Achilles underneath the laptops. The circuit boards are attached by Velcro to enable easy manipulation of circuit location and accessibility.

5.1 Electrical Innovations

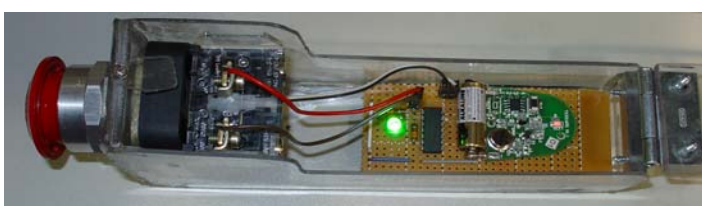
All the circuit boards were manufactured using a lithography process. The circuit is designed using a PCB layout using computer aided software. The circuit is then printed out on a transparency which is used to shield the board from the light source. The board is then exposed to light and then processed in ferric chloride to dissolve the unprotected copper. All of this is performed in a dark room and is a very inexpensive method to make circuit boards quickly. This innovative technique gives Achilles more professional and reliable electronics components.

5.2 E-Stop

The manual Emergency-Stop (E-Stop) is mounted on the top of the robot chassis. It is required safety control device necessary for an autonomous robot. Additionally, there is a wireless E-Stop, which serves the same purpose but can be utilized 500ft from Achilles, ten times greater than the minimum distance required. The wireless E-Stop works on a duty cycle of 75% which means that the process can continue, and shuts off when the cycle is reduced to 25%.

Before the signal is processed, it is sent through a low pass filter. The signal transmitted is a basic pulse train, with either 75 or 25% duty cycle. In order for the signal to be understood, the average voltage of the pulse-width modulated signal is taken. If the transmitter has a high voltage value of 5 Volts, the average for a 75% duty cycle will be 3.75 Volts. For a 25% duty cycle, the average voltage will become 1.25 Volts. This average is then drawn into a comparator circuit with a 2.5 Volts comparing voltage. This differentiates between the two duty cycles, and the output will result in either a high, 5 Volt for 75% duty cycle or a low, 0 Volt for 25% duty cycle. This either keeps Achilles running or shuts it down.

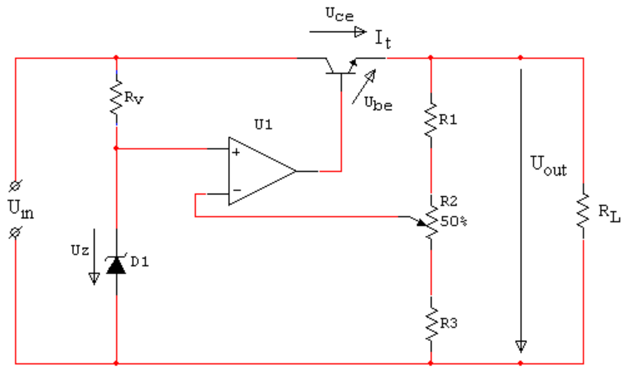
Once the wireless transmitter (see figure below) has sent the disabling signal, Achilles cannot be reactivated wirelessly. To restart it, the manual E-Stop must be turned off and then on. This is a safety precaution to prevent Achilles from reactivating when it shouldn't.



5.3 Battery Charger and Power

Achilles has 4 laptops and 2 12V batteries that need to be charged during down time. For quick and easy recharging of these components Achilles can plugged in to a normal 120V AC outlet. This allows all of the components to stay in the robot while being charged.

A step down transformer was lowers the voltage from 120V AC to about 22 RMS. After the transformation, a diode bridge is used to complete the process, acting as a DC rectifier. Because the input voltage recharging the lap tops must be very stable, a large capacitor is connected in parallel with the load, minimizing the effective ripple voltage.



The voltage created, U_{in}, is drawn into a stabilizing circuit, left figure. For R1 and R3, potentiometers of 10K and 1K were used, respectively for rough and fine tunings. R1 and R3 were used to maintain a constant and stable load voltage of 19 V, regardless of load resistance. A different load voltage is used for recharging the 12 volt batteries, so the potentiometer settings are varied in order to obtain an optimal charge voltage.

This table displays the power output of Achilles:

Type of Battery	Amp Hours	Weight	Device Controlled	Duration (Hours)
PS-1212 12V Lead Acid Battery	1.3	1.32 lbs	Fans/electronics	1.3
PS12120 12V Lead Acid battery	12	8.50 lbs	Hub	4.6
PS-12260 12V Lead Acid Battery	26	18.0 lbs	Motors	0.87

5.4 Motion Control

The vehicle uses two 12 volt DC motors which are operated using PWM motor controllers. The motors will be connected to the motor controllers using Amp King 8 gauge wires. The motor controllers that are being used are the Victor 883s which have a rating of 24 volts at 60 amps of continuous current. They are very reliable and are considerably less expensive than most other controllers with the same ratings and interface. The motor controllers will receive a 5 volt PWM signal from the microcontroller. The speed will be controlled by adjusting the duty cycle on the PWM which will in turn adjust the speed of the motors proportionally.

5.5 Digital Compass

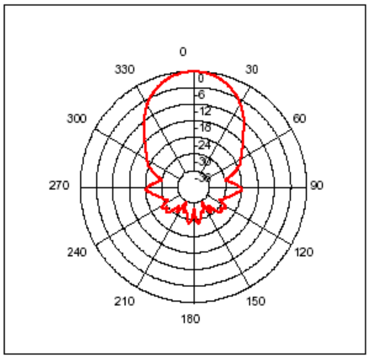
Turns will be controlled with the assistance of a digital compass. The compass we will be using is a Devantech CMPS03 with 0.1 degree of resolution and 3 to 4 degrees of accuracy. The direction of the compass will help determine which wheel should spin faster or slower in order to be turned in the desired direction.

5.6 Robostix

The main objective of the Robostix is to communicate to the sensor devices using Two-wire Serial Interface and analog-to-digital converters. The Robostix gathers the data and sends it to the ARM9 board using UART. In addition, the Robostix is physically running the robot by generating PWM signals to the motor controllers, which directly run the motors. The PWM operation has its own external crystal clock to improve the efficiency of the process by avoiding the interruptions that would occur when doing operations such as I2C.

5.7 ARM9

The Robostix uses C programming. The program starts by initializing the PWM to neutral at 11% duty cycle to avoid any interference to the motor controller. Then the program sends a general call to the I2C devices and verifies that everything is responding and working properly. After making sure that all the sensors are working, the program runs a loop waiting for instructions from the laptop or the ARM9 board. When the robot starts running, the Robostix will gather the data from the tachometer in every loop. In addition, for safety purposes, the ARM9 board and the laptops send a heartbeat signal, which serves to make sure that the laptops and the ARM9 board are working properly and the program is still running.



5.8 Sensors

Connected to the Robostix board is an ultrasonic range finder sensor that can identify objects between 3 centimeters and up to 6 meters away (see figure below). The range that is measured can be in inches, centimeters, or microseconds. The range finder uses about 12 milliamps during a range find. The figure on the right shows the beam pattern of the range finder.

A circuit that senses any loss of power preemptively warns the Robostix of impending failure and the microcontroller then shuts down the motors to prevent the vehicle from running haywire.

6.0 Software design

Robotic software is often times complicated and computationally intensive. For both of these reasons, a divide and conquer design philosophy in which each individual problem is solved in a separate module was implemented for the robot. To speed development of a rather complicated series of components, existing programs and frameworks were adapted to suit the needs of the robot. The algorithms used to solve each part of the problem were researched thoroughly and planned to ensure seamless integration and accomplishment all of the required tasks.

6.1 Software Innovation

The most significant innovations to Achilles come in the software portion. There are multiple software components and techniques that give Achilles an edge over other robots.

Stereo vision enables Achilles to see further and get a better understanding of what obstacles lie ahead. No laser range finder could do what stereo vision can do. The types of obstacles Achilles can see are drop offs, switch backs, and past obstacles.

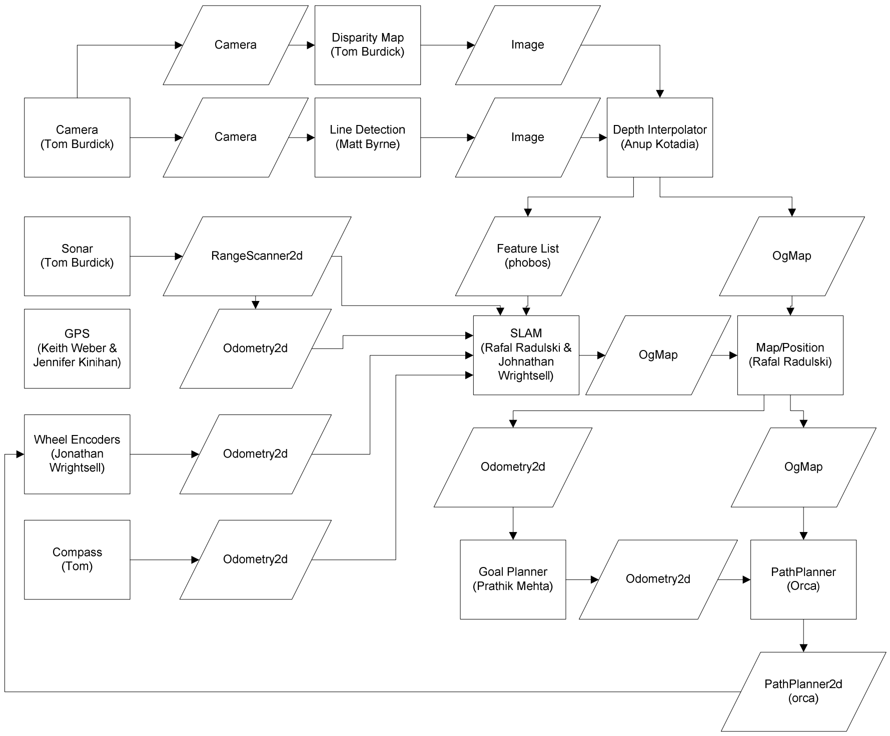
The graphics processor creates the disparity map. This technique provides an implementation 4 times faster than the next competitor. Having the quick disparity map allows for better reaction times.

Distributed computing enables Achilles to do more things and do them faster and more robustly. Most robots have one laptop computing everything. Distributing the computing time between many computers makes Achilles much faster.

Look-ahead path planning is used with a probabilistic understanding of location of Achilles and obstacles (SLAM). Achilles does not just obliviously follow the lines and avoid obstacles along the way. It creates a map of its surroundings and intelligently choses the best path to take.

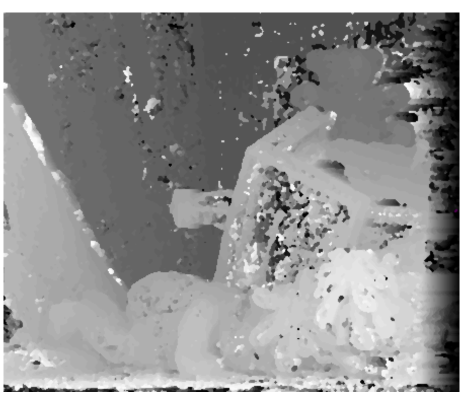
An innovative framework to avoid duplication of effort called Orca is implemented. Orca was written by universities in Berkeley, CA and Sydney Australia specifically for the DARPA Urban Challenge. Achilles uses some of their framework and some of their components which implement well known and tested algorithms such as VFH+, Wavefront Propogation, and Skeleton path planning.

6.2 System Integration



6.3 Stereo Vision Disparity and Recovering the Z value

In the field of machine vision it is a common problem to be able to take what a camera sees and translate it to world coordinates. A single camera standing still does not provide enough information to recover the full world coordinates of what it is seeing. Stereo vision was chosen for its ability to recover world coordinates without movement. It was also chosen for its ability to be used in conjunction with line detection to map lines in 3-D. This additional information could then later be used to preplan paths far in advance.



In stereo vision there is a left and right cameras are aligned such that they lie on the same horizontal plane, view the same direction, and are parallel. The physical distance between the centers of the two cameras is used in the transform. Each image is treated as a two dimensional matrix of three values; red, green, and blue. The upper left most corner is designated the index (0,0). The bottom right most corner is the largest coordinate value in the image. A point in the world will then be mapped in to the two images differently. There will be a horizontal distance in pixels that the point in the world is shown in the left and right images. This distance is known as disparity. By calculating the distance for each point in the image a disparity map can be formed. This is a 2-D matrix, or image, that contains integer values with the disparity at that particular pixel. It can be displayed as a grayscale image as shown above. White color pixels are close and gray color pixels are farther away.

The cameras themselves do not perfectly conform to the perspective transform by default. The perspective transform assumes a pinhole camera is being used and thus no lens distortion. Lens distortion, or barrel distortion as it is sometimes known, causes straight edges in the world to appear curved. Correcting for the distortion to conform to the needed pinhole assumption is called rectification of the images. A commercial stereo camera was used that came pre-calibrated with easy to use C function calls to rectify the images. Once the distortion is gone correlation between pixels in the left and right images (shown above) must be done to determine which pixel in the left image matches a pixel in the right image. Usually this is done by taking the absolute difference of intensity of each pixel along a row in one image with respect to a single pixel in the other. This creates a 3-D matrix of absolute difference values from which a minimum value may be determined. The minimum value is determined for each x,y coordinate in the matrix, the z index which contains the minimum value at the point is the disparity. The two images will never necessarily contain the same exact value at a matching mapped point due to noise and variation in lighting. There might also be numerous pixels in the same row which contain the same value as the one being checked. Either way an incorrect disparity choice for a given pixel is easily made. To correct for this error several steps are taken.

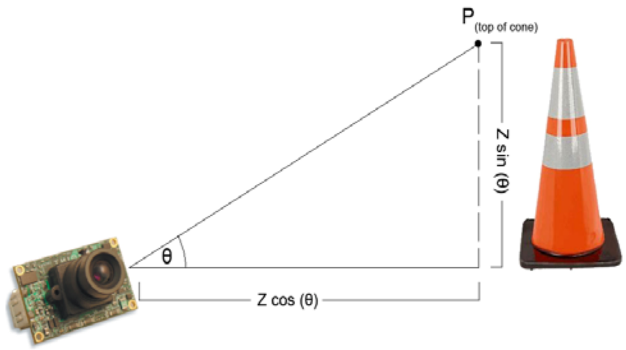
First a weight window is applied at each pixel to determine the difference. So instead of just taking a single difference value at a single pixel, several neighboring pixel differences are weighted in on and summed together to form a new difference value for a given point. This software uses a Gaussian mask of 5x5 to determine the difference value for a given point.

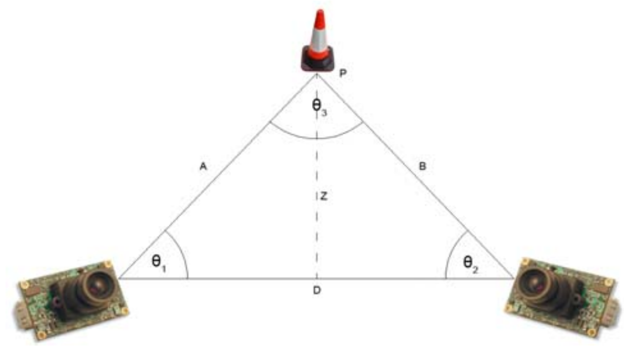
Second a 5x5 median filter is applied to the disparity map image. This eliminates stray values, what is known as salt and pepper noise. This creates a very smooth grayscale image that can then be used to determine the z values for each point.

The number of calculations to develop a disparity map adds up very quick with even relatively small images of 320 by 240 pixels in dimension and checking for 32 disparity values. The graphics card is therefore used to dramatically speed up these operations. By using a modern day graphics card to do the image processing, in this case the Stereo Vision component achieves about 30 frames per second, or 30 Hz. A similar implementation done on the systems normal x86 processor from the commercial vendor who supplied the camera was only able to achieve around 8 frames per second, or 8 Hz.

6.4 Position Interpolation

This is the first step in a three step process to convert two 2-D images into a usable 3-D map. The parallax technique is used to find the position in space of each particle picked up by the cameras. Parallax is the same





technique used by astronomers to find the distance to stars. The use of two cameras allows Achilles to perform this task.

The color of a pixel in the disparity map encodes the difference between what the two cameras see. A black pixel encodes zero difference between the two camera images and a white pixel represents the maximum possible difference between the two images. Each pixel of the disparity map corresponds to a point seen by Achilles in real space. The angles (see figure at left) at which a point appears in relation to each of the cameras are calculated from the disparity map, as are the angle of elevation of the point based on the color of the corresponding pixel for that point. Based on this information and using the distance between the cameras as a baseline, the location of each point can be determined in 3-D space relative to the vehicle using the law of sines. Every possible input for each pixel location is then calculated ahead of time and stored in a lookup table to ensure that the program runs at the maximum possible speed.

6.5 Object Association

After Position Interpolation is done then Object Association must be performed. This is the second step in the 3-D map process that determines whether an object exists in the world it sees. The method used divides the Disparity Map into sections and determines the similarity between adjacent sections.

A quad tree data structure divides the Disparity Map. This method divides the disparity map into small square sections of pixels, and calculates the statistical mode of the pixel values in that subsection. All the pixels in one section are then assigned the value of the mode. The quad tree marks those pixels as belonging to a single object. However, if a section does not have a certain amount of pixels being equal to the mode, the section will be ignored since that signifies that there is too much noise to make an accurate reading.

The algorithm compares the mode of the current section to adjacent section modes and if there are matching adjacent modes, the matching sections are grouped into a single object. This area is further compared to adjacent sections to create bigger objects. The process then continues until either, no more adjacent sections can be added to the current object or a certain threshold is reached. The object associations are then sent to the object-mapping module of the software.

6.6 Object Mapping

The final step is Object Mapping. This component creates an overhead view of the 3-D map just created by the Position Interpolation. A bird's eye view of the course ahead, provides a base for the Path Finding component.

The objects identified in the previous steps are converted to rectangular approximations. This is accomplished by taking the four corner pixels from the disparity map for each detected object in the quad tree, and using their locations and encoded values from the lookup table. The four converted locations contain information that is used to determine the smallest rectangle that can surround the specified object. The rectangle is then placed in a data structure representing the area of the course occupied by the object.

The other objects in the scene are all placed on the map data structure in the same fashion and then the map structure is sent to the SLAM component. The lines are handled differently since it is not easy to classify lines as simple rectangles, especially if the lines are at an angle relative to the robot. The line detection output indicates which pixels of the disparity map correspond to lines. The software then marks every pixel in the disparity map, which is indicated to be a line as a line fragment. Fragment positions are determined using the lookup table and the locations are marked on the overhead map. A new overhead map is then also submitted to the SLAM component for further processing.

6.7 SLAM

Simultaneous localization and mapping (SLAM) creates an accurate map of obstacles within a course and improves localization of the robot. Localization error creates mapping error and likewise for mapping. The SLAM component attempts to solve the two problems using the technique of FastSLAM.

SLAM receives data from object detection, wheel encoders, GPS and compass. The Object Detection component provides a list of observed features, including obstacles. The objects are 3-D structures, described in terms of dimensions and location relative to the robot. Wheel encoders, GPS and compass relay information to SLAM about the new position of the robot. SLAM periodically creates a limited size 2-D map centered on the current robot position. It also continuously outputs Achilles' position that is most consistent with its internal data.

The main element of SLAM is a particle filter, were each particle is a map. The particle filter is responsible for managing the maps. Each map is a list of all known features, with their descriptions, and position of Achilles. At each iteration, maps are updated with a new position of Achilles. However, true position of the robot is never known. The information from wheel encoders, GPS, and compass contain some unknown error. To compensate for the error, each map is updated with coordinates that are randomly, slightly modified. Therefore, some of the maps are updated with coordinates that are closer to the correct values.

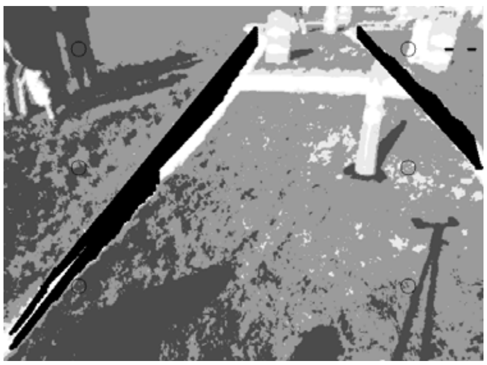
Once coordinates of all maps are updated, they receive from object detection a list of features currently seen and recognized by the vision component. The new information is merged with existing data. This is done independently for each map.

Each map records the new observation differently because they have different assumed coordinates of Achilles. If wrong coordinates are used multiple times, the errors aggregate. After some number of iterations, there will be important differences between maps. Some maps will better represent the world, while other will contain large amount of errors. Therefore, each map will have a quality to it. The quality is calculated based on new observations. If the new observations match well to already acquired data, the quality of the map is high; if deviations are high, the quality is low. At the end of an iteration, the particle of highest quality is chosen, and coordinates of Achilles stored at that map are output. These coordinates represent the most probable position of Achilles.

6.8 Line Detection

The Line Detection component discriminates course borders from the rest of the image. This component will send an IPL grayscale image to the Position Interpolator component. The image consists of black and varying scales of gray. The black (0,0,0) tells the depth interpolator where a line has been found. White and gray are not used outside of the Line Detection component.

Before writing an algorithm, the properties which were most unique to the lines on the course were considered. The lines are mostly vertical with some change in angle. On average they are white, and the image can be manipulated to accent that quality further. To do this, a segmentation algorithm is implemented to turn the image into a grayscale image. The result is also an image whose contrast is increased, and therefore the difference between the line and the grass beside it is more apparent.



To find a point on the line, three horizons on the image are considered. To allow for debugging, these horizontal lines are one-third of the image apart and can be offset either up or down. Six points (circles in figure at left) are traced from the left and right on the horizons. They move towards the center until they approach what is approximated to be a white line. Those points then allow for the Line Detection component to trace up and down the line. If the line curves, the spaces on either side are averaged. All of the black is sent to the depth interpolator, and while some objects outside of the course will be included as lines, SLAM is designed to take care of these anomalies and ignore them.

Breaks in the lines are handled by connecting the tops of the bottom points with the bottom of the middle points; the fact that the points move from out to in allow for errors to be directed outward. If errors were directed inward, Achilles might be confused by the appearance of lines in its path.

6.9 Drive Control

Achilles' drive train is controlled using two Robotics Connection Hamamatsu P5587 Wheel Encoder Modules. The wheel encoders detect the velocity and direction of each individual wheel. Velocity and direction are picked up from the two 44 spoke black and white encoder disks

The information picked up from the encoder disks is transferred by the wheel encoders via a pulse width modulated (PWM) signal to the microcontroller. The microcontroller communicates between the software environment and the motor controllers via a serial connection to a selected host Laptop. Every 100ms a sign-magnitude byte value that directly dictates the velocity of a wheel is sent to the laptop. The laptop uses this information with the other components of the software and then relays a new byte to the microcontroller. The microcontroller uses a look-up table to send the correct PWM signal to the motor controller.

The software component Drive Control uses a proportional integral derivative (PID) control algorithm for precise control based on a target velocity. This algorithm takes, as input, a target velocity ["setpoint"] and the current velocity of the wheel ["process-variable"]. The output is then the sum of the PID transfer function that is multiplied by the gain. Gain values are chosen for a fast rise time, minimal overshoot, and minimal steady-state error.

7.0 Vehicle Cost

2007-2008 Intelligent Ground Vehicle Competition (IGVC)
Budget Report Summary

Qnty.	Component	Retail Price(ea.)	Sub Total	Discounted (ea.)	Discounted Total	Supplier
Power Electronics						
2	12V 26AH Batteries	$ 38.80	$ 77.60	$ 10.00	$ 20.00	Battery.Web.com
2	12V 12AH Batteries	$ 21.76	$ 43.52	$ 5.00	$ 5.00	Battery.Web.com
1	12V 1.3 AH Batteries	$ 11.12	$ 22.24	$ 1.00	$ 1.00	Battery.Web.com
2	24V Victor 833 M. Controller	$ 149.00	$ 298.00	$ 149.00	$ 149.00	IFI Robotics
Computer Components and Electronics						
1	PointGrey Bubblebee2Camera	$ 2000.00	$ 2000.00	$ 0.00	$ 2000.00	Point Grey
1	GPS Hemisohere	$ 1500.00	$ 1500.00	$ 1500.00	$ 0.00	Jknelectronics
1	Arm9 board	$ 53.00	$ 53.00	$ 53.00	$ 53.00	Technologic Systems
1	Robostix	$ 49.00	$ 49.00	$ 49.00	$ 49.00	Gumstix.com
1	Gumstix connex 200xm	$ 109.00	$ 109.00	$ 109.00	$ 109.00	Gumstix.com
1	USB 12-Channel GPS receiver	$ 34.99	$ 34.99	$ 34.99	$ 34.99	Amazon.com
2	ASUS Laptop	$ 1200.00	$ 2400.00	$ 1200.00	$ 2400.00	Newegg.com
1	ASUS Laptop	$ 1300.00	$ 1300.00	$ 0.00	$ 0.00	Newegg.com
1	HP Laptop	$ 150.00	$ 150.00	$ 150.00	$ 150.00	CDW
1	6 PORT 1394 HUB	$ 40.00	$ 40.00	$ 40.00	$ 40.00	Frys.com
1	5 PORT Network Switch	$ 27.99	$ 27.99	$ 27.99	$ 27.99	Newegg.com
1	2 Channel RF Relay	$ 39.49	$ 39.49	$ 39.49	$ 39.49	HobbyTron.com
4	SRFO8 Ultrasonics Sensor	$ 54.11	$ 216.44	$ 54.11	$ 216.44	RobotShop
1	Magnetic Compass	$ 57.99	$ 57.99	$ 57.99	$ 57.99	RobotShop
1	Wheel Encoder Module (pair)	$ 36.99	$ 36.99	$ 36.99	$ 36.99	RobotShop
1	Spoke Encoder Disk (pair)	$ 17.99	$ 17.99	$ 17.99	$ 17.99	RobotShop
Chases and Hardware						
5	Low-Carbon Square Steel	$ 34.40	$ 172.00	$ 34.40	$ 172.00	McMaster-Carr
4	Aluminum L-Channel	$ 19.19	$ 76.76	$ 19.19	$ 76.76	McMaster-Carr
2	Aluminum Stock (1/8 thick)	$ 74.31	$ 148.62	$ 74.31	$ 148.62	McMaster-Carr
16	Latches	$ 5.36	$ 85.76	$ 5.36	$ 85.76	McMaster-Carr
3	Polycarbonate Sheets	$ 61.18	$ 183.54	$ 61.18	$ 183.54	McMaster-Carr
1	Button Screws (100 pack)	$ 11.06	$ 11.06	$ 11.06	$ 11.06	McMaster-Carr
1	Hardened Precision Shaft	$ 18.44	$ 18.44	$ 18.44	$ 18.44	McMaster-Carr
1	Sprockets and Chain	$ 37.87	$ 37.87	$ 37.87	$ 37.87	McMaster-Carr
4	Air Linear Bearings	$ 7.40	$ 29.60	$ 0.00	$ 0.00	McMaster-Carr
1	8” Pneumatic Caster Wheel	$ 61.96	$ 61.96	$ 0.00	$ 0.00	Gilmore-Kramer
2	12.5” Wheels	$ 34.99	$ 69.98	$ 0.00	$ 0.00	Robot Mkt. Pl.
2	12V Motors	$ 317.00	$ 317.00	$ 0.00	$ 0.00	Palmer Ind.

Total Cost	**Retail Cost $ 9,786.83**	**Discounted Cost $ 6,490.93**

8.0 Conclusion

Approximately 3000 man hours were invested in Achilles for this year. Relying on stereo vision sets us apart from other teams as being a very cheap and effective method for self navigation and vehicle design is rugged, sturdy and easy to manufacture. Achilles is using four laptop computers to communicate to the various sensors to work as efficiently and as fast as possible. The entire process of building this vehicle was a very beneficial learning experience for all the team members. Each member was exposed to the other disciplines of engineering and could appreciate each others dedication. The talents and skills of each team member were utilized and work was completed. Care was taken to ensure that future members will have the knowledge necessary to improve Achilles and to further engineering knowledge of autonomous navigation.

Special thanks to UIC faculty, Sponsors, and the UIC College of Engineering for their support in this project.