NorMAN

Northridge's Mobile Autonomous Navigator

<u>Team Members</u>
Fast, Nathan
Hurtado, Manuel
Limon, Gerardo
Pablo, Froylan
Petrosyan, Avetis
Torres, Levis

<u>Alumni Advisors</u>
Weaver, Andrew
Ubowski, Mike

Required Faculty Advisor Statement:

I certify that the engineering design of the vehicle described in this report, NorMAN, has been significant, and that each team member has earned four semester hours of senior design credit for their work on this project.

C. T. Lin
Department of Mechanical Engineering
Cal State University, Northridge

1. Introduction

California State University Northridge is proud to present Northridge's Mobile Autonomous Navigator (NorMAN). NorMAN will be competing in the 16th annual Intelligent Ground Vehicle Competition (IGVC) in May 2008. The NorMAN team is composed of multi-disciplinary students with the goal of designing a state of the art autonomous robot. We have undergone many design improvements and are now ready to compete in the IGVC.

1.1 Design Innovations

From the beginning of the year, one of our primary goals was to develop a vehicle with innovative algorithms, features, and technologies. The primary power system is comprised of a hydrogen PEM fuel cell battery hybrid. This hybrid power system increased the run time by more than 50% over last year's robot. The power consumption profile and remaining power reserves of hydrogen and battery of the hybrid power system are monitored in real time and displayed, alerting the user of a low power condition. The cognition system uses Kalman filtering of several sensors to get an incredibly accurate robot position and heading. Local and global probabilistic obstacle maps are constructed from Laser Range Finder (LRF) and line data. Our cubic spline algorithm uses this mapping to generate a path for both the autonomous and navigation challenge. An Inertial Measurement Unit (IMU) is used to decrease the GPS localization standard deviation by more than a factor of 20 when compared with the raw GPS information. The vision system uses particle analysis, line probability mapping, and a line continuity algorithm in its filtering process to extract accurate line data and generate path goals. The data from each sensor is fed into the cognition hub using our LabVIEW shared variable engine to enable parallel processing and optimal sensor refresh rates. Custom shaft adaptor couplers joining the gear boxes to the wheel hubs were designed and machined that enabled the reduction of the vehicle width. The innovative ideas, algorithms, and products that have been generated and integrated into this robot by this team have poised NorMAN to win the Intelligent Ground Vehicle Competition.

1.2 Team Structure

This years' team was based on a chain of command structure. The team leader helps facilitate communication with the different groups as well as keep the project as a whole in perspective and on track. An organizational chart is shown in Figure 1.

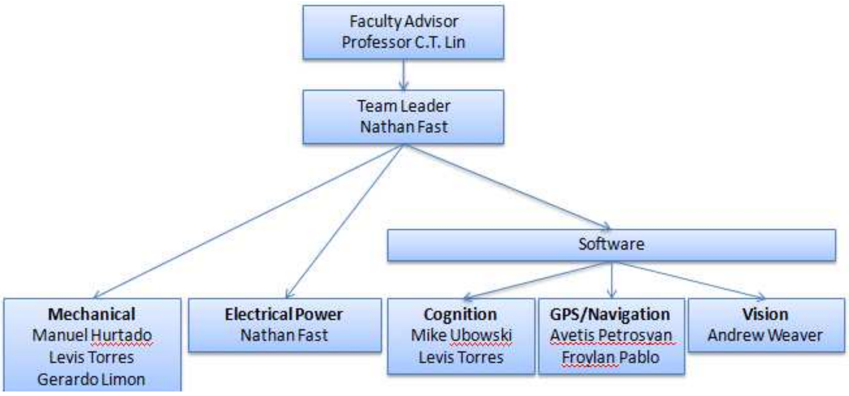

Figure 1. Team organization.

2. Mechanical

2.1 Design

NorMAN was designed with the goal of weight reduction and distribution, as well as the ability to pass through a conventional door. This design has a safety factor of 2, which is made to support an evenly distributed weight of 700lbs. Care was taken to place the weight of the drive train and the computer on the rear of the vehicle close to the wheel axle to aid in vehicle maneuverability. The power source is placed on the bottom front of the vehicle to lower the center of gravity improving stability.

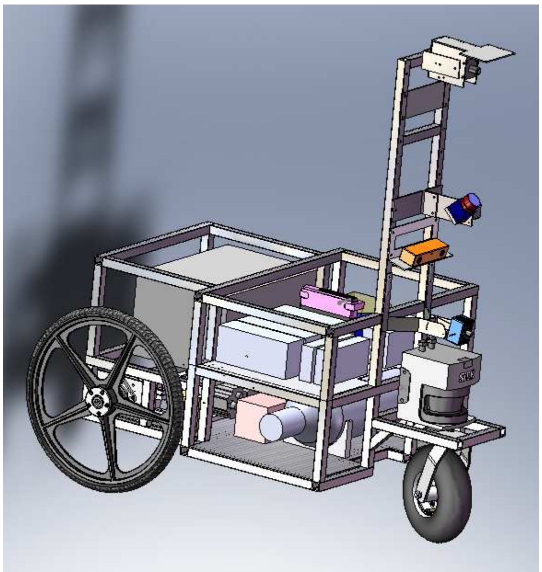

Figure 2. SolidWorks rendering of NorMAN.

2.2 Chassis

The chassis of NorMAN is constructed of 1in sq. 6061 Aluminum tubing, welded together and heat treated to maintain 6061 properties. 6061 aluminum was used due to its high strength to weight ratio, and its compatibility with the onboard components. The square tubing was selected for structural support, as well as the ability to mount the shell directly on the chassis. The ability of NorMAN to fit through a 32” door was a key issue when designing the chassis. NorMAN measures, 54” long, 28” hub to hub, 52.5” at the top of the mast, with a ground clearance of 6.5 inches. Another key design is a collapsible mast required for transportation of the vehicle. The chassis is also equipped with a sliding tray for easy removal and replacement of the power source.

2.3 Wheel Configuration

The wheel configuration is partly influenced by our goal of size reduction. A differential-drive three-wheel design was chosen due to its ease of controlling the vehicle direction and stability. This design has two 20 inch utility wheels with built in keyway hubs as drive wheels on the back and one 10 inch pneumatic castor wheel in the front.

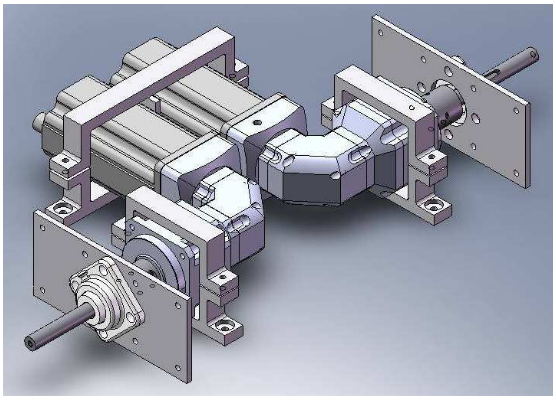

Figure 3. SolidWorks rendering of motor/gearbox configuration.

2.4 Drive Train

NorMAN is driven by two Quick Silver QCI-A34HC-2 motors connected to right angle Apex Dynamics gearbox ABR090 with special NEMA 34 flange mount with a 30:1 gear ratio. The right angle gear boxes and custom designed shaft adapters greatly reduced the width of the vehicle. We designed and machined our own shaft adapters to combine the coupler and shaft into one. Due to the component’s placement of our drive train it was determined that the majority of the vehicle’s weight and stress in shear, would be placed on the coupler shaft. The coupler shaft is made of plain steel and machined to intake the output shaft of the reducer and step down to output the size of the wheel hub. The step also serves the purpose of holding the wheel in place. The motor mount and reducer mounts were also designed, stress tested, and machined in house to brace the drive train for the least possible movement and vibration. Rubber padding was affixed on the interior of the mounts to absorb a good portion of vibration and impact to the drive train.

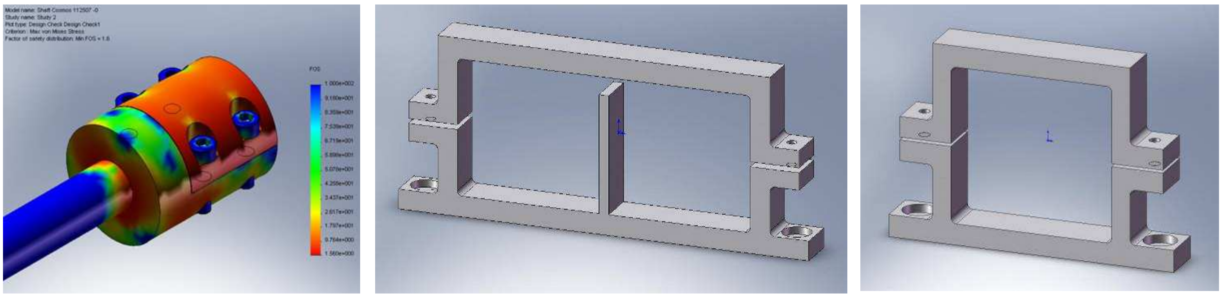

Figure 4. SolidWorks rendering of coupler shaft (left) and motor mounts (middle and right).

2.5 Shell

The chassis was purposely designed to have an integrated shell. The shell is constructed of 6061 aluminum sheeting and plexi glass, all of which is mounted to the chassis. All panels of the shell are either bolted or hinged to allow for easy access to components. The shell panels are gray, black, and red flat colors to reduce glare to the cameras and to match our school colors. NorMAN is painted in a digital camouflage design to keep with the theme of unmanned robotics.

3. Electrical

The Electrical Power system is comprised of several subsystems. These are the DC power distribution system, the two interchangeable power sources, and the power monitoring system. An integration of all three of these subsystems are innovative products that has not been offered on previous robots. The power distribution is implemented with a professional, reliable printed circuit board (PCB) designed for high power, individualized subsystem power paths, and quick disconnect capability. The power source is comprised of either a hydrogen fuel cell battery hybrid system or a lead acid battery system. The power monitoring system monitors current and voltage from the fuel cells and batteries, calculates the estimated hydrogen tank capacity and battery capacity, and warns the user when the power resources are low.

3.1 Power Distribution

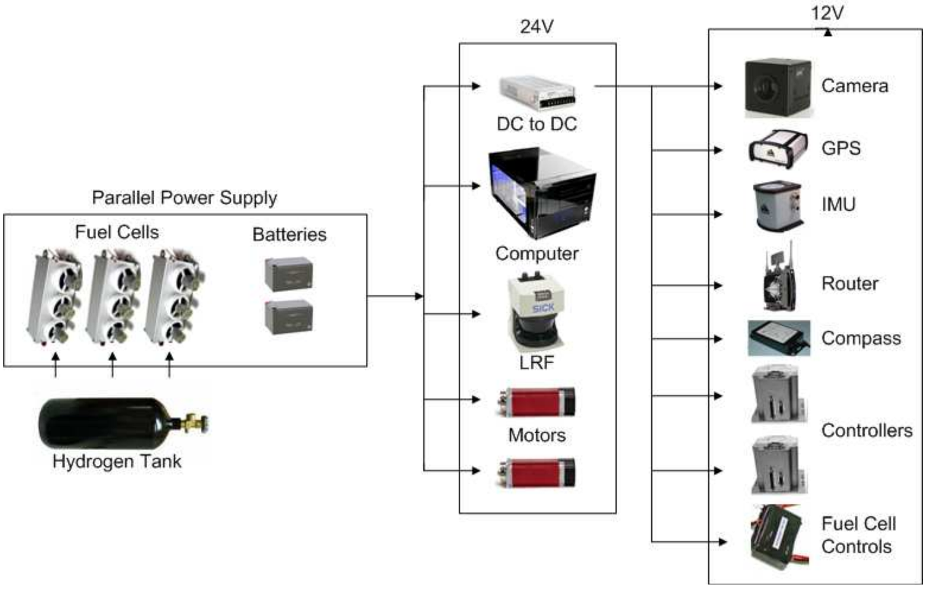

Figure 5. Overview of PCB power distribution system.

NorMAN features a PCB power distribution system shown in Figure 5. The equipment and sensors used on the robot utilize both 24V and 12V power. The 24V power is supplied by the Fuel Cell – Battery Hybrid source or the battery pack and the 12V power is supplied with an on-board DC-DC converter that is routed back into the PCB for centralized distribution. This PCB distribution panel is a multilayered board that was designed for high power use. It utilizes high current capable Anderson connectors for ease of assembly and quick disconnects capability. Each subsystem is separately fused and has a dedicated switch. This provides for localized troubleshooting and maintenance of the power systems and reduces unnecessary power consumption during subsystem testing. The PCB also has an integrated port dedicated to power monitoring. Current sensors and voltage traces were designed into the board and their signals are routed to a PCB mounted DB-9 connector. These signals are then taken from the board to the data acquisition (DAQ) unit to monitor the power consumption of the robot.

3.2 Fuel Cell Battery Hybrid Power System

Two modular interchangeable power systems were designed for NorMAN. The first is two 12V 28Ahr lead acid batteries in series. The second and primary power system is a parallel Fuel Cell – Battery Hybrid system. Two modular power systems were implemented on NorMAN to allow for parallel testing and integration of systems on the robot. This allowed software and sensors to be continually tested using the battery power system while the hybrid power system was being tested and integrated separately.

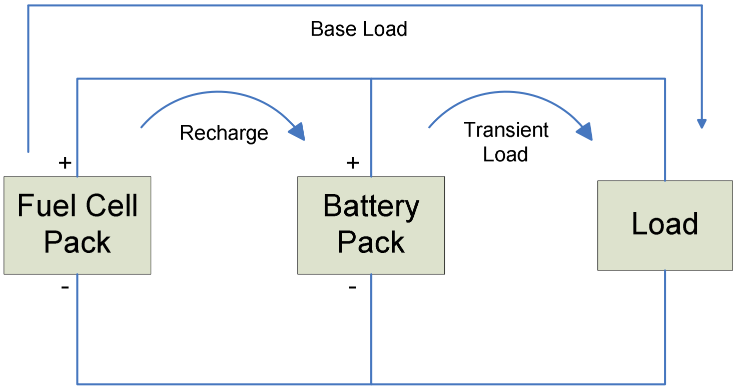

Figure 7. Fuel Cell Battery Hybrid system.

The system level operation of the Fuel Cell Battery Hybrid is shown in Figure 6. Three Horizon H-300 hydrogen fuel cells are connected in parallel with two 12V 12Ahr lead acid batteries. Each fuel cell has a DC-DC converter integrated into the controller pack which is 88% efficient and is set for 27.5V. The fuel cell pack provides the base load to all of the equipment that is always on, such as the computer, LRF, camera, etc. The battery pack handles the surge of transient power that is drawn from the motors. When the motors are off, the fuel cells recharge the battery pack. The worst case scenario base load is 619W. After the DC-DC losses, the fuel cells provide 792W and therefore can provide a maximum of 6.3A at 27.5V to recharge the batteries. This particular voltage for the fuel cell DC-DC converters was chosen because it was within the voltage variation specifications for all of the 24V equipment, it was within the maximum float voltage for the batteries in series, and is high enough to cause a potential difference with the batteries to charge them during motor idle. With the 12Ahr batteries, the motors can run at top speed for 24 minutes continuously before needing to be recharged by the fuel cell pack.

	Device	**Max Power (Watts)**
Max Base Load	Computer	350.0
	Router	12.0
	Laser Range Finder	50.0
	GPS Receiver	2.5
	IMU	16.0
	Camera	4.0
	Compass	0.2
	Motor Controllers	48.0
	Fuel Cell Controllers	126.0
	Fans	9.5
	E-Stop	0.4
	Total Max Base Load	**618.6**
Max Transient Load	Motors	907.0
	Worst Case Scenario Load	**1525.6**

Figure 6. Worst-case power consumption calculations.

The hybrid power system was designed with future expansion capabilities in mind. Currently, three 300W fuel cells and one 29 cubic foot tank of hydrogen are used. The NorMAN chassis was designed to fit up to four 300W fuel cells and two 29 cubic foot tanks comfortably. This allows for future customers and applications of NorMAN to have a higher maximum power capacity and twice the run time of the current configuration.

3.3 Power Monitoring

NorMAN is equipped with a powerful, easy to use power monitoring system. The current and voltage of the battery pack and fuel cells are measured using a USB DAQ. This data, along with the sampling rate is fed into a LabVIEW algorithm that calculates the amp-hours that are either being sourced or sinked by the batteries during discharge or recharge. The current and voltage measured from the fuel cells is used in conjunction with the specified fuel consumption rate and the DC-DC conversion efficiency to estimate the number of standard liters of hydrogen remaining in the tank. The remaining battery amp-hours, remaining standard liters of hydrogen, raw current and voltage data, and overall power consumption trend is displayed in real time to the user. This information is also data-logged for post-processing trend analysis. This allows for the future study and optimization of the dynamic hybrid operation of the power system. The data-logging function also allows the user to initialize the power reserves to the last known state before system shutdown occurred.

4. Computer & Sensors

4.1 Computer and System Integration

The Robotic Systems Mission Computer (RSMC) is the new mission computer onboard NorMAN. It is fully designed from the ground up meeting all specifications to handle the computational tasks and communication requirements with the NorMAN sensors. The parts are cheaper and faster to replace than the previous year's proprietary PXI because their widely available off the shelf. In addition, RSMC runs an Intel Quad Core 2.4GHz processor with 4GB of 800MHz DDR2 memory which computes many more tasks than a conventional single core processor. To handle all the sensors onboard the NorMAN, the RSMC is equipped with a special serial card which contains 8 customizable ports at customizable speeds. Each port can be set to RS-232, RS-422, and/or RS-485 with any baud rate ranging from 50bps up to 1.8432Mbps depending on the serial communication used in addition to its 6 USB ports, a FireWire port, and an Ethernet port for all the other sensors. RSMC incorporates a DC power supply which eliminates the need for an AC to DC converter onboard the NorMAN. The end result is a robotics systems mission computer enclosed in a mini tower case that conserves space, weight, and handles the required computations with minimal effort. Figure 8 represents the current system integration of the NorMAN sensors communicating with the RSMC.

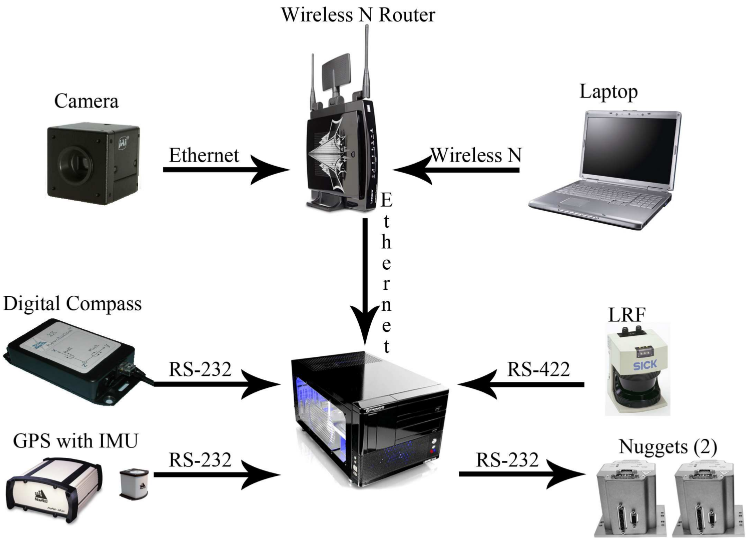

Figure 8. RSMC and System Integration.

4.3 Laser Range Finder


Figure 9. Laser range finder.

NorMAN's main source of obstacle detection is a SICK LMS291-SO4 laser rangefinder. This device is capable of scanning a range of 180º in 0.25º increments, measuring distances up to 80m away. The settings used for NorMAN make the device scan a range of 180º in 1º increments, measuring distances up to 8m away and returning values in mm. Any distance further than 8m will not be considered by the obstacle avoidance algorithm, and 1º increments are sufficient at this distance. An RS-422 serial interface was used in order to obtain a data transfer rate of 500kbaud.

4.4 GPS/IMU


Figure 10. GPS receiver (left) and IMU (right).

NorMAN is incorporating the Novatel SPAN System which consists of the ProPak-V3 GPS Receiver along with an LN200 Inertial Measurement Unit (IMU). Using a serial communication of RS-232 with the RSMC, a baud rate of 57600 and log data rate at a maximum of 20Hz is achieved. The Log BESTPOSA gives the best possible computed solution using both the GPS and the IMU. The data received from the SPAN System consists of latitude, longitude, and azimuth and is converted from latitude and longitude to x and y in Cartesian coordinates.

4.5 Camera


Figure 11. JAI camera.

The camera used is a JAI A70GE. The camera has several attractive features, such as selectable 8-bit or 10-bit color output, a ½" color CCD, Gigabit Ethernet output, 4-12mm Tamron lens with Auto-Iris functionality, and an angle of view greater than 90 degrees horizontal. Maximum resolution is 766x572 at a frame rate of 60 FPS.

4.6 Digital Compass


Figure 12. True North 2X Revolution compass.

The compass that is being using for NorMAN is the 2X Revolution by True North. This device has an accuracy of 0.3 degrees and can be updated at a frequency of 12Hz. It communicates using an RS-232 serial communication interface for data transmission.

4.7 Motor Controller

The Silver Nugget N3 M-Grade controller/driver has a 10-bit ADC for single input from a signal range of 0 to +5 VDC. The controllers have a RS-232 serial interface and baud rate of 57.6k with a variety of preset commands for controlling and interfacing. The controller communicates with the encoder by pulse width modulation at a TTL logic level of +5VDC. The encoder provides 16000 counts/rev with a maximum rotational speed of 4000RPM.

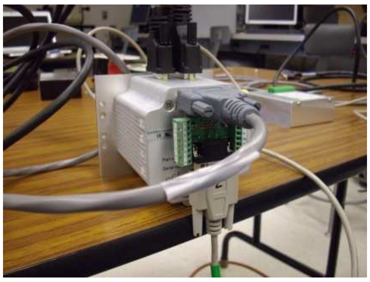

Figure 13. Silver Nugget N3 M-Grade controller/driver.

5. System Software

5.1 System Communication

The software used to program NorMAN was entirely written in LabView 8.2. This GUI application was very user friendly when interfacing all our sensors together. By using the embedded functions that LabView provided, sending and receiving commands from our outside sources became quite simple.

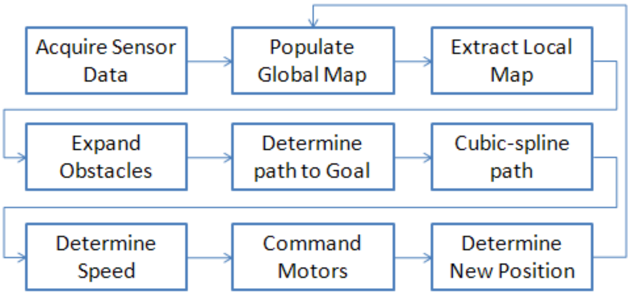

Figure 14. Software system.

5.2 Vision

The goal of vision is to provide cognition with locations of white lines and a general heading for the robot to move towards. This is accomplished by a continuous sequence: Acquiring vision data from a camera, filtering to accent white lines, analyzing the image to remove noise and extract white lines, and examining line data to determine goal heading.

The vision code has been written for portability so it can be moved to any platform and used as a viable vision solution. This program, called the “Vision Shell”, allows the same piece of code to be tested through several mediums (videos, still images, or live camera input) with minimal additional programming.

Because cognition needs a birds-eye view of the obstacle course, the camera image must be corrected for both perspective and lens distortion. A grid of evenly spaced markers is laid in front of the camera, as seen in Figure 15, to shows the extent of the perspective and lens distortion. Corrective parameters from this image are extracted and saved as a specific camera profile to be applied back in the vision process.

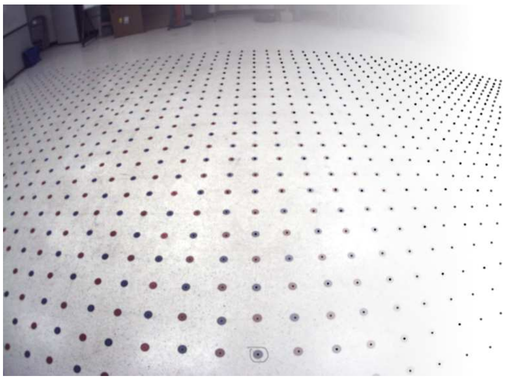

Figure 15. Camera calibration. Image fades from camera photo to digitally edited image for calibration.

Vision data is filtered by way of channel mixing, where the green and blue channels are extracted from the camera input and the inverse of the green channel subtracted from the blue channel. The resulting grayscale image is then passed on to particle analysis, which groups adjacent white pixels as particles and generates several properties of each particle such as center of mass, bounding rectangle, and area. Given that lines are generally long and not compact in form, the set of particles retrieved from the analysis is further filtered by extracting only particles whose properties match the characteristics of a line. Remaining particles are further filtered by passing it through our custom line continuity algorithm that classifies all particles into sets describing lines with the criteria that particles in a set should be within maximum distances and orientations thresholds. The longest sets are considered to be the real lines. The output particles are corrected for perspective and lens distortion using information obtained from a calibration image of the camera and the results are passed on to vision mapping. The process is shown in Figure 16.

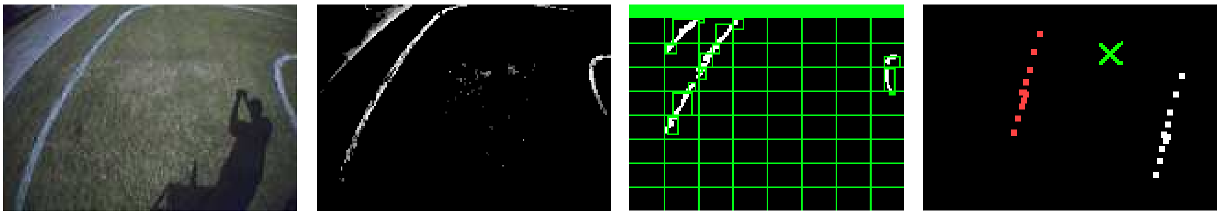

Figure 16. Vision filtering process. From left to right, original image, filtering, particle analysis, and line continuity. In fourth image, line continuity rejects the short lines and replicates the strong left line with an offset. Green X is calculated goal.

Once filtering is complete, the result is passed to the vision map, which takes additional inputs from the compass, GPS, and motor dead-reckoning to orient and position local vision data onto a global probabilistic map. Instances of both the obstacles and the local area seen by vision are kept and used to calculate the probability an observed obstacle both exists and is not noise. Each pair of cells in the obstacle and area map can be unique and thus a dynamic probability threshold is achieved even with a more or less static probability filtering threshold. This final output is passed to cognition as a vision map along with the computed goal from line continuity. Figure 17 displays some test results.

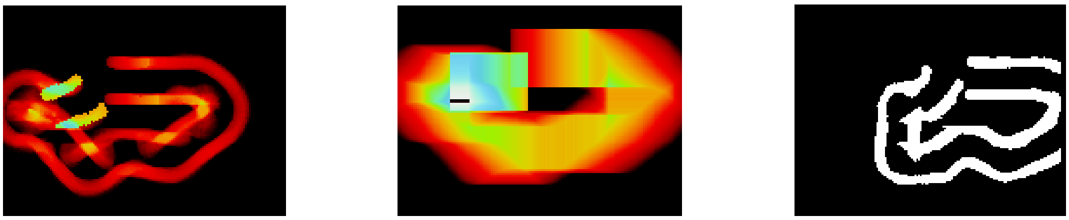

Figure 17. Probability mapping test run output. Left window showing obstacles seen; middle, the area seen; and right, the output.

The advantage of the particle analysis over the previous year's Hough-Line transform analysis is accuracy and speed. The Hough-Line is a best-fit formula and can easily fail and wildly oscillate from small amounts of noise. Due to the nature of the algorithm, a single white pixel implied a large set of possible lines that had to be calculated. This caused a significant slow-down that the particle analysis easily trumps. The time saved from particle analysis allowed this year's vision algorithm to incorporate more complex filtering to achieve greater and more accurate vision results.

5.3 Obstacle Avoidance, Mapping, and Path Planning

NorMAN uses a laser range finder to build a Cartesian style occupancy grid map around itself. The laser range finder outputs a 181 array of distance points at 20Hz. The points are converted into XY points and used to populate the grid map. There are two different style maps that NorMAN uses. A local map and a global occupancy map, shown in Figure 18.

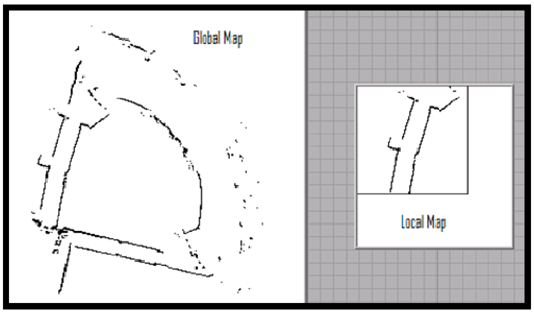

Figure 18. Two maps used for localization.

The Global map is fixed at a coarse scale to fit the entire map in an N x M array. This array is large enough to fit the entire map but coarse enough to not waste system memory. Once the immediate area around the robot is mapped and saved in the global map, a 16 x 16 m² area is extracted and used in a local reference. This map is what is used to perform our immediate moves. The obstacle and white line points are expanded by an offset equal to the radius of the robot. This is done because the robot sees itself on the map as a dimensionless point. This new array is then fed back to the path-planning program. Filled cells are obstacles and unfilled cells are clear. This map is used to determine incremental steps towards the desired occupancy grid goals. With the use of parallel processing, we can designate one process to solely building the global map and another process to extracting the local map and using a path planning algorithm called Grass Fire. Occupancy grid goal cells are usually located on the edge of the map, pointing to the next waypoint relative to NorMAN or the goal point found by the vision system during the autonomous challenge, with NorMAN being at the center. The grass fire algorithm fills in empty cells with numbers representing the distance to the goal until all of the cells around NorMAN are filled. Then a path finding algorithm outputs a cell to cell path based on the information provided in the grass fire algorithm as shown in Figure 19.

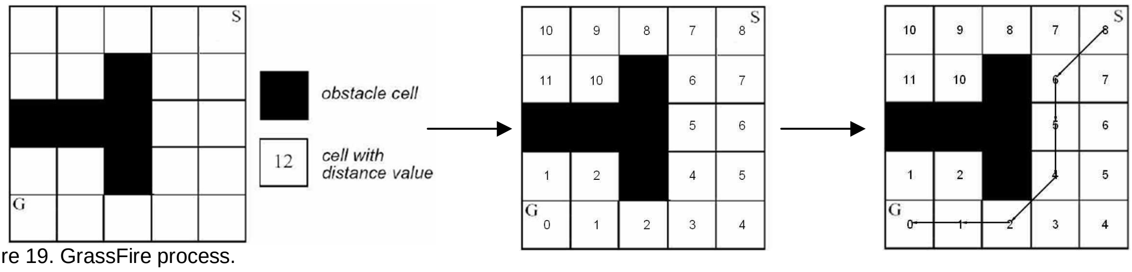

Figure 19. GrassFire process.

The output cell array is then converted to discrete x and y coordinates. These discrete coordinates represent the incremental steps towards the desired occupancy grid goals. Since there are parallel processers computing the path and extracting data, NorMAN will continuously create new paths. With the new found path to the goal,

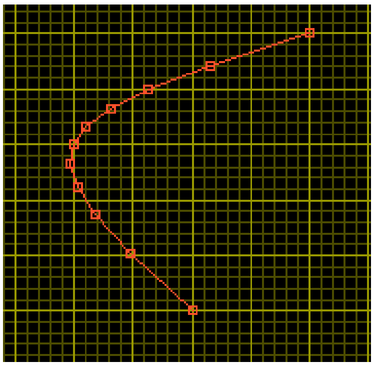
Figure 20. Cubic spline of a GrassFire path.

a filter needs to be implemented to smooth the trajectory. To do so, a cubic-spline algorithm, Figure 20, is implemented.

5.4 Autonomous

The Autonomous Challenge and Navigation Challenge both use the same core cognition algorithm with a few modifications. To accomplish this, NorMAN must first gather all information from each sensor. This is done with parallel processing which ensures that the data is received from each sensor at their maximum rate. The key sensors used in the Autonomous Challenge are the LRF and the digital camera. Both of these sets of data are fused into one local map and placed in a larger, coarser global map. This map contains the entire competition course. Our local referenced goal is generated by the vision algorithm's analysis of the line data.

NorMAN computes its immediate position using odometry but will deviate substantially over long distances due to losses of contact of the wheel and the ground. To correct this, we Kalman filter the odometry position with the GPS / IMU position. This corrects the inaccuracy of the long distance while maintaining the accuracy at short ones. NorMAN also computes its' heading using the same odometry but once again, over long distances, it distorts. The same filter is applied with the digital compass and IMU for this correction.

5.5 Navigation

Motion Control for the Navigation challenge was coded similar to the Autonomous Challenge in efforts to make the algorithm compatible with both challenges. Instead of using the goal from vision, GPS will run in a parallel process to compute the global goal in reference with the lat / long position represented in Cartesian coordinates. The Cartesian coordinates are converted from latitude and longitude which are fed into motion control to plan its' path to the next navigation point while avoiding obstacles. Due to an average standard deviation of 7 meters using just the GPS solution, minimization of error is required to perform well during the navigation challenge. Therefore, OmniSTAR HP is used to decrease the solution error down to an average standard deviation of 0.7 meters using just the GPS coordinates. We can improve this even better while the IMU. This in conjunction with the GPS gives us an average standard deviation of 0.3 meters.

Azimuth is supplied from the GPS receiver and compared to the digital compass data to give a more precise heading. This helps reduce the error in the digital compass caused by magnetic interference. Since the global goal is outside of the local map, a new goal is generated with the same heading as the global goal until the global goal is inside of the local map. At that point, the same procedure occurs as mentioned before.

5.6 JAUS

Joint Architecture for Unmanned Systems was a new system that we incorporated on the NorMAN. A great learning curve was required to understand the material. Numerous documentations provided from www.jauswg.org were read through to aid in the development of the new JAUS software along with learning about UDP packet transmissions through a network. In addition, the IGVC JAUS rules were also of great help by making some parameters constant, although the most difficult process was learning about the correct values in the JAUS packet.

JAUS was developed using LabVIEW, being built upon basic UDP transmission. An OCU in addition to the NorMAN subsystem node was developed to send and receive JAUS messages across the ,same network. This would verify, the messages being sent and received. The parameters are set by the user in the OCU which is a GUI application to reduce time creating different JAUS messages. The OCU takes the user defined parameters and creates a JAUS UDP packet in ASCII hexadecimal representation, the packets are then sent by the OCU across the same network to the destination IP address specified for the subsystem NorMAN on port 3794. The subsystem NorMAN receives the JAUS message and parses it. Various parameters in the header section are compared to the values specified on the IGVC JAUS rules for validity. If the message complies with the parameters set forth by the IGVC JAUS rules and the JAUS architecture documentations on www.jauswg.org. The message data portion will then be compared to the ASCII equivalent of “JAUS01.0” to verify the message data. If all requirements are met as of yet, it will check the command code to determine what type of command the OCU desires. The subsystem NorMAN node then composes a JAUS packet and sends it back to the OCU specifying if the JAUS packet it received from the OCU was a valid JAUS message or not. Level 1 JAUS compliance is completed because the current software does not take action from the command code. A representation of our JAUS architecture is represented in Figure 21. Level 2 and Level 3 will be implemented in the following year to come.

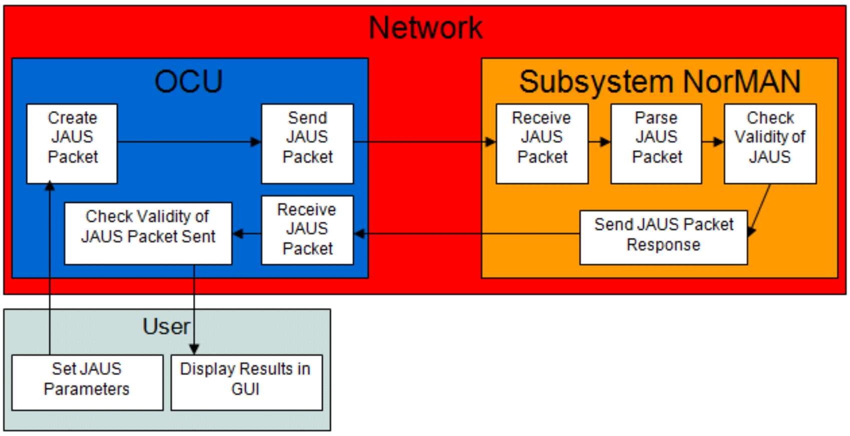

Figure 21. Our implementation of JAUS architecture.

Many challenges were faced during the development of the JAUS software. One such challenge involved the nature of UDP packets not requiring a handshake unlike TCP packets. As a result UDP packets are transmitted much faster but at a cost of being dropped. This became an issue in our implementation of our JAUS software. Developing the OCU to send a UDP packet across the network, it would wait for a reply from the NorMAN subsystem node. When a UDP packet was dropped, the OCU would freeze in to an infinite loop waiting for the message to be received. To account for this problem, a new check system was implemented in the development of our JAUS architecture by waiting 25ms for a response back from the NorMAN subsystem node. If the message was not received by this time, it would stop waiting and send another request. This process would be continued for a total of three continuous requests. If the OCU did not receive a message from the subsystem NorMAN after three continuous requests, then the OCU would stop. If a message is received in between the additional requests, then the process is reset to zero. Through many trials, this method has proven successful with a maximum drop of two continuous packets. Another problem arose from choosing a type of JAUS architecture that would work efficiently. Many trial architectures were proposed and implemented but our current architecture proved to be the best since it has not failed yet. The process was a challenge because of the creation of our own JAUS architecture but as a result it gave us a much better understanding of JAUS.

6. Predicted Vehicle Performance

6.1 Reaction Time

With parallel processing and shared variables, NorMAN now computes each process separately without the need to wait for any other process to finish. This allows it to process each sensor input and execute manipulation VIs at its maximum potential.

Reaction Time	
Process	Time (ms)
Sensor Input	110
Grass Fire	440
Map Building	140

Figure 22. Reaction time of sensors and cognition algorithms.

6.2 Speed

NorMAN's speed is based off an equation that involves the distance from its current position to its immediate goal. The speed is bounded with a maximum speed of 3 mph and minimum of 0.5 mph for safe traveling and course navigation.

6.3 Ramp Climbing Ability

The difference of the average robot speed on a 15 degree slope to the average speed on flat ground is about 0.001 MPH. A slope has little effect on the robot when it is traveling at a constant speed over a ramp.

6.4 Run Time and Battery Life

Assuming that the motors are running 20% of the time and all of the equipment experiences their worst case scenario power consumption, the fuel cell battery hybrid system is calculated to provide 79 minutes of run time with one 29 standard cubic foot tank of hydrogen instead of the 52 minute run time that the 28Ahr batteries alone would provide under the same conditions. This hybrid power system achieves a 52% longer run time over the previous year's system.

6.5 Sensor Range

The LRF has a 180 degree sweep with a maximum range of eight meters. With the Vision code and the JAI camera, the most accurate range that can be seen is about five feet ahead. The GPS and the compass operate best outdoors.

6.6 Complex Obstacles

If the robot finds its self heading towards a dead end, a local goal will be placed on the other side of it. Grass Fire will fill find the shortest path to that goal point which is back from where it came from. Once the robot turns around, a new goal will be drawn in front of it and now be heading away from the dead end. Center Islands are viewed as any other obstacle. Depending on the robots current position a trajectory will be computed around the center island. If the island happens to be perfectly centered in front of the robot so that both paths are equidistant, they are equally likely to be chosen. Potholes can be easily identified with the particle analysis in the vision code.

6.7 Accuracy of Arrival to Navigation Waypoints

The accuracy of the robot's arrival to navigation waypoints is limited by the standard deviation of the GPS, which ranges from 7m (free service), 0.6m (OmniSTAR), to 0.3m (IMU integration).

7. Safety

When creating any autonomous vehicle it is very important to take safety into account. A careful safety analysis is essential to ensure that no one will be harmed while operating, observing, or working on the vehicle and has been incorporated into NorMAN.

7.1 E-Stop

NorMAN's Emergency Stop system has two methods of immediately stopping the vehicle: wireless and hard-stop. If either system is activated, it will send a direct 5 volt source into the breakout modules of the controllers to safely shut down the motors. The wireless portion includes keyless entry with a handheld transmitter and receiver with a whip antenna that sends a signal within a range of 50 feet. The wired portion uses a large red 120V 10A regulated switch that controls the on/off states of the 5 volt regulator. The on/off states are indicated by logic light emitting diodes on the E-Stop enclosure on the mast.

7.2 Hydrogen Safety

Several layers of safety were designed into the hydrogen fuel distribution system on NorMAN. The first layer of safety is Containment. This consists of the hydrogen tank, 2 stage regulator, fuel distribution manifold, connectors, and tubing. These components form the primary system of containment to prevent a hydrogen leak. The second safety layer consists of several design features in NorMAN that will each reduce the probability of a catastrophic accident in the unlikely event of a hydrogen leak. This includes proper ventilation, heat isolation, and rubberized insulation of the chassis. There are two small ventilation fans located at the top of each of the two compartments where hydrogen is used. If there is a leak, the lighter than air hydrogen will rise to the top of the compartment and be vented outside, preventing a buildup of hydrogen inside the vehicle. The hydrogen tank is isolated from heat producing systems such as the computer, electronics, and fuel cells. Keeping the tank cool helps to maintain a safe internal tank pressure. The last safety design feature is the insulation of the compartments where hydrogen is used. The inside of the fuel cell and tank compartments were coated with several layers of clear, rubberized paint to reduce the probability of sparks or arcing near the hydrogen equipment.

7.3 Chassis Ground

Due to the aluminum frame and shell, it was very important to properly ground the chassis of NorMAN. This will protect users from electric shock in the case of a hot wire coming into contact with the chassis. If this were to happen, the energy discharge would be routed safely to battery ground instead of through the user to ground. The chassis was grounded at the power distribution PCB using 6 AWG wire to ensure a low resistance path to ground.

8. Cost

Although NorMAN is classified as an R&D vehicle, every effort was made to reduce costs through sponsorship agreements with material and component suppliers. This was done within the constraints of time and the effort to provide an innovative product. Figure 23 shows the team costs and retail costs associated with NorMAN.

Item:	Team Cost:	Retail Cost:
Mechanical		
Motors (2), Controllers (2), Encoders (2)	$1,790	$4,400
Gearbox (2)	$2,174	$4,348
Wheels (2)	$73	$118
Hardware	$407	$407
Raw Material	$123	$1,173
Electrical		
Fuel Cells	$9,271	$10,272
Hydrogen Tanks (8)	$396	$792
Fuel Distribution	$18	$54
Hydrogen 2 stage regulator	$203	$483
12Ahr Batteries (4)	$101	$276
28Ahr Batteries (4)	$320	$608
DC-DC converter	$104	$104
E-stop	$93	$93
PCB	$550	$550
Electrical Components	$560	$560
Data Acquisition	Donated	$129
Computer & Sensors		
Computer	$2,225	$2,225
LRF	Used	$7,675
Camera and Lens	$1,115	$1,300
Digital Compass	$397	$467
GPS Receiver, Antenna, IMU Enclosure	$8,500	$28,079
Inertia Measurement Unit (IMU)	Loan	$44,000
Total:	**$28,420**	**$108,113**

Figure 23. Team costs and retail costs of NorMAN.

9. Conclusion

NorMAN is an autonomous ground vehicle that was designed, built, and tested by the students of the CSUN IGV Team and integrates innovative algorithms, features, and technologies into one vehicle. The fuel cell hybrid power source, our use of positional Kalman filtering, probabilistic obstacle mapping, vision line continuity algorithm, and shared variable engine set us apart from the competition and will help establish new standards in the field of autonomous robotics.