Submitted to:

AUVS Ground Robotics Competition
Design Judging Panel

by:

Nithin Chandrababu
PavanVempaty
Praveen Samala
Priya Samala
Nadeya Shafee

Department of Electrical and Systems Engineering
School of Engineering and Computer Science
Oakland University
Rochester, MI 48309-4401

May 24, 2006

ABSTRACT:

The dynamics and control of an unmanned vehicle using vision and laser range finder for control was modeled and simulated to drive through a test course. The vehicle chassis is a modified lawn mower, with two independently controlled drive wheels, which control both velocity and steering coupled together and a caster wheel in the front. The drive motors are controlled by an MPC 555 microcontroller to achieve the commanded velocity at each wheel. With the capability of controlling both wheel velocities, the global speed and steering of the vehicle can be controlled with a proportional controller or a fuzzy controller. To verify these dynamic models and controller models, simulations were performed incorporating forward and reverse vehicle kinematics that was developed. The vehicle was then constructed to accommodate the laser range finder and tested. Test results agree with simulation results. The vehicle is capable of navigating an outdoor track while avoiding obstacles of different colors.

Table of Contents

Acknowledgments

Thanks to Prof. Ka C. Cheok, our team advisor and mentor for the numerous ideas, encouragement, step by step guidance, without him we wouldn't have been able to compete. Our gratitude also to Jadi, Inc. for letting us borrow equipment and to Edzko Smid, Michel Franken, Alex Kurienko, Leonard Brown, Forrest Wright, Frank Cox, Bill Young and many others for their valuable advises, time, and support.

Introduction

In preparation for the AUVSI Ground Vehicle Competition, we have compared classical control methods and fuzzy logic as options for the vehicle navigation scheme. To assist in the comparison, vehicle simulations (using Matlab) were used to draw conclusions about each control method to reduce the time required for design and testing. This project's goal is to understand the vision processing, dynamic modeling and control aspects of the vehicle using simulations, and to validate those simulations by building and testing the hardware. Our vehicle is called X-MAN, to signify "unmanned" system.

Problem Statement

Given a course consisting of an outdoor grassy terrain with possible inclines, sandy regions, and shadows, the vehicle must perceive its upcoming environment through the camera and laser sensors, and drive around the course once without leaving the lane boundaries. The second part of the course consists of navigating from a starting point to a number of target destinations (waypoints or landmarks) and return to base or start.

System Overview

Figure contains X-MAN's System Integration Diagram. X-MAN uses two sensing systems: vision and laser range finder. Information about the environment is obtained from these sensors and processed by the PC. The PC decides on an appropriate course of action, and sends the desired wheel speeds to the Speed Controller.

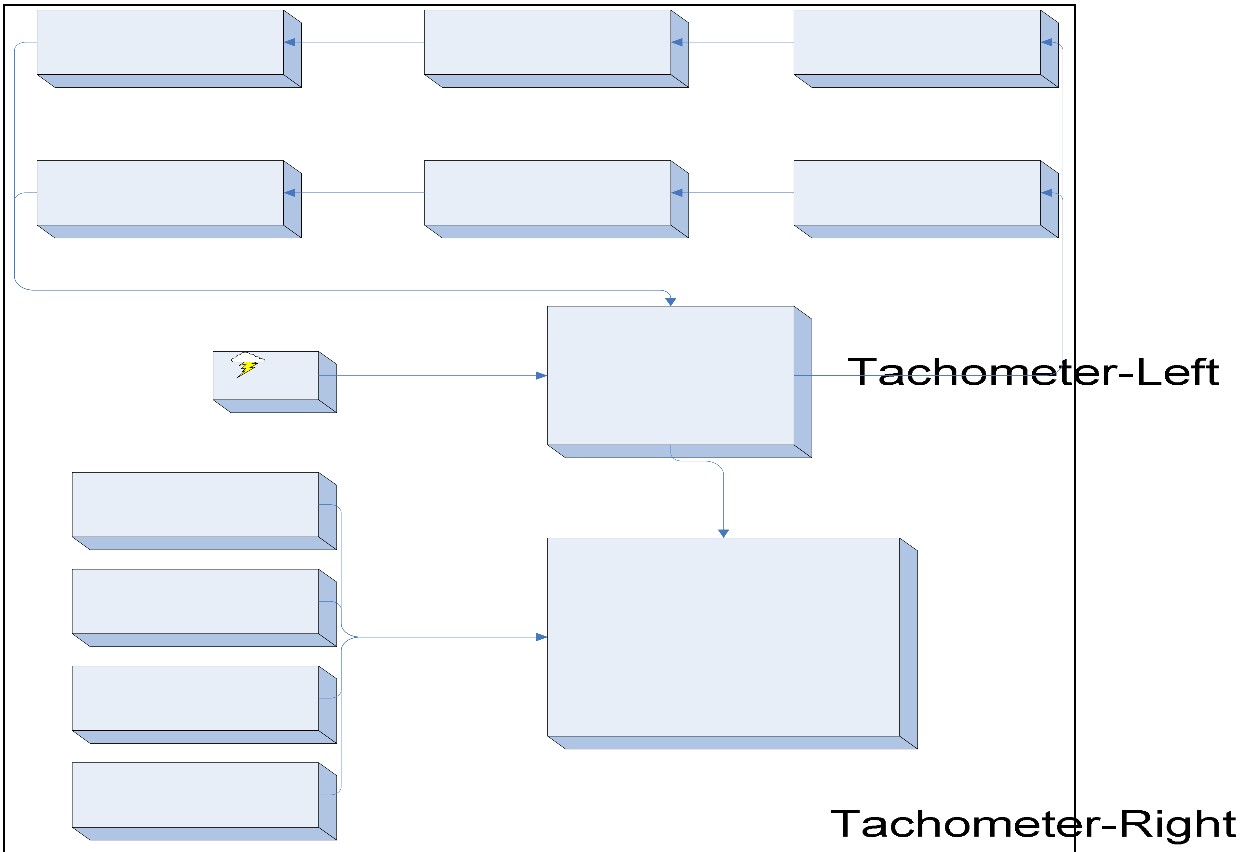

Figure 1—System Integration Diagram

Vision

This is the primary sensory method for the autonomous challenge event. Images are obtained with a webcam. Entire image processing is done with the PC (refer to Image Processing section). Vision provides information about both lines and obstacles. Line and obstacle detection is based on color only, without regard to texture or shape.

Laser Range Finder

This is the secondary sensory method for the autonomous challenge event. This consists of a SICK LRF(laser range finder) mounted in front of the vehicle. SICK LRF offers accurate distance measurement of 80 meters throughout a scanning field upto 180 degrees. The PC mounted on the vehicle is responsible for the operation and data collection from the sensor. The range information received from the SICK LRF is used to determine the position of obstacles in the vehicle internal map.

GPS

This is the primary sensor for the navigation challenge. The GPS modus used is an usb based Delorme GPS. The PC mounted on the vehicle reads in the GPS coordinates and converts the same into a local East-North-Up (ENU) frame.

Chassis

X-MAN is built upon an Friendly Robotics RoboMower. The robotic mower is designed to support around 100 pounds at up to 4 mph. Through using independently controlled rear drive wheels, speed and steering are realized as two aspects of the same function. This speed/steering arrangement is a central piece of the X-MAN operating theory.

Power Distribution

The power distribution system installed on the X-MAN has the following features:

1. Ease of use. The number of subsystems can grow without it becoming difficult to connect them to power. Standardized connectors increase the ease with which a subsystem can be removed and reinstalled on the vehicle.
2. Two electrical buses, 12 volt and 24 volt are used in the vehicle. This enables us to use flexibility as far as voltages are concerned.
3. Safety. In case of a failure it is necessary to quickly disconnect power from all subsystems. The amount of current flowing from the batteries during operation is inconvenient (if not dangerous) to pass through a hand operated switch. Therefore, a fuse is used in both the buses.
4. Reliability and Appearance. The number, current carrying capability and routing of all conductors is carefully considered. The connectors and terminals are sufficiently rated for the tasks they perform.
5. Expandability. Extra terminals are included to account for future needs.

Electronics

The heart of the X-MAN is an MPC 555 microcontroller. The microcontroller A/Ds reads in the analog values of the steer and speed signals from a joystick in manual mode or the D/As of a DAQ(1208FS) connected to the onboard PC in autonomous mode and converts it into appropriate right and left wheel speed signals which are fed into LMD18200 H-Bridge circuitry to drive the left and right wheel motors. Hall effect sensors feeds back the wheel speeds for precise control of each drive wheel.

Main Controller

High-level path planning, object detection and control is done using an onboard Pentium computer with hyper-threading technology. It integrates all sensor information, plans a path, and determines what vehicle and steer speeds are necessary to accomplish that path. It then passes the speed information to the MPC555 microcontroller. Two high-level control schemes were evaluated: proportional control and fuzzy control.

Emergency Stop (E-Stop)

When activated, this system shuts down the vehicle's motor driver circuit. The remotely controlled E-Stop button was implemented using a commonly available garage door opener at the microcontroller level. For high reliability and safety, the manual E-stop is implemented completely in hardware. This assures that a computer bug or system crash cannot prevent the manual E-stop system from functioning.

Safety, Reliability, and Durability Considerations

X-MAN is a safe, reliable, durable vehicle for several reasons. First, the vehicle cannot physically attain a speed greater than 4mph, a limitation carried over from the original design of the electric Robomower.

System reliability and durability were important considerations throughout the design process. The electric Robomower chassis is sturdily constructed, and motors are designed to survive years of use and abuse; this system is not a converted child's toy. The motors and control hardware are intended to service a normal house lawn for several years. Also, all subsystems built by the team are constructed with ruggedness and reliability in mind.

Vehicle Control Strategy

Path Boundaries

From an autonomous vehicles point of view, the obstacles and lines represent the same kind of entity. That is, they are both *boundaries* which must not be crossed. All lines, dashed or solid, and all obstacles, no matter what size or color, form boundaries between which the vehicle must choose a path. Figure 2 illustrates this concept.

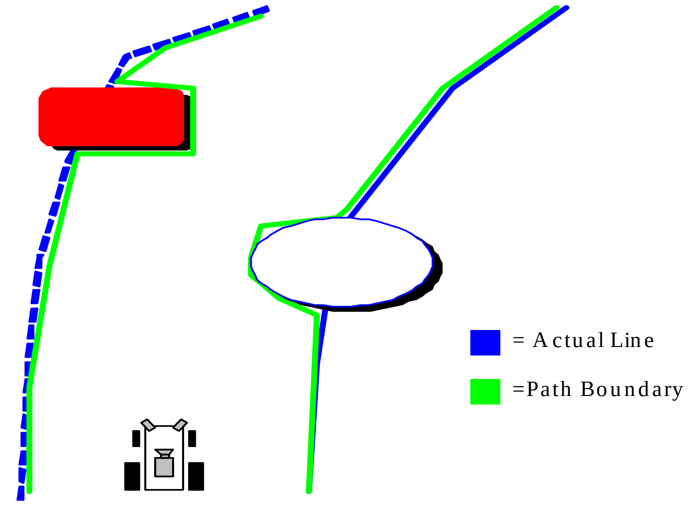

Figure 2—Lines, Obstacles, and Boundaries

The sensors in the system report boundary points to the *path planner*. At this time, only vision and LMR exist on the vehicle. However, this scheme works well to integrate new sensors. The only requirement of the sensor is that its results can be represented as a set of boundary points. The boundary points are divided into two sets: the left boundary and the right boundary. Whenever a point is reported from a sensor, it is specified as either a left or right boundary point. A complete set of boundary points might look like that shown in Figure 3.

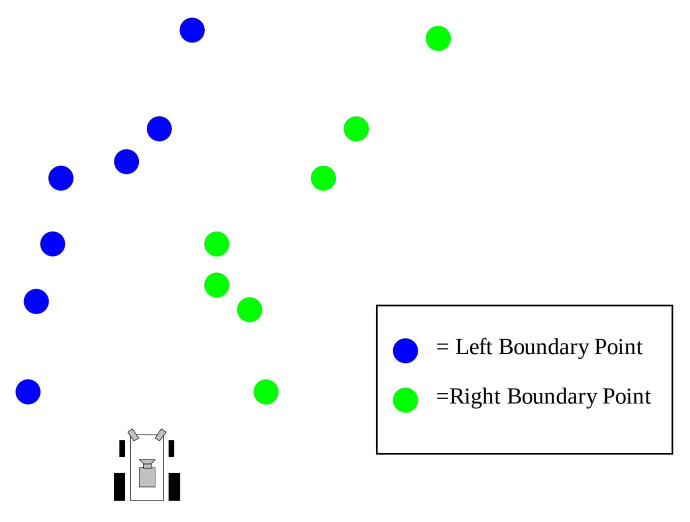

Figure 3—Left and Right Boundary Points from Sensors

Vehicle Kinematics

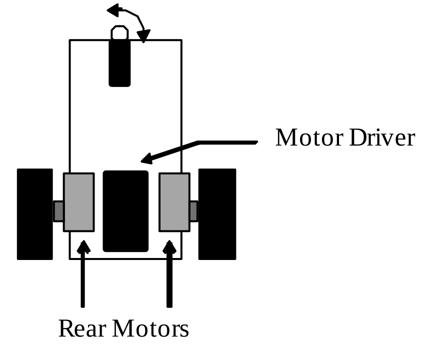

Figure 4--Vehicle Top View

The vehicle chassis is an electric Robomower, with independently controlled drive wheels, which control both velocity and steering coupled together. The vehicle's rear wheels are driven independently by two electric motors. When there exists a differential speed between the left and right wheels, they will twist the vehicle, producing a turn (steering). No explicit steering system is required. The front wheel is free to rotate in any direction, as shown in Figure 4, thus supporting the front of the vehicle while allowing the rear wheels to steer. A diagram of the vehicle executing a typical turn is shown below in Figure 5. This type of vehicle exhibits the special property of the speed and steering systems being coupled together.

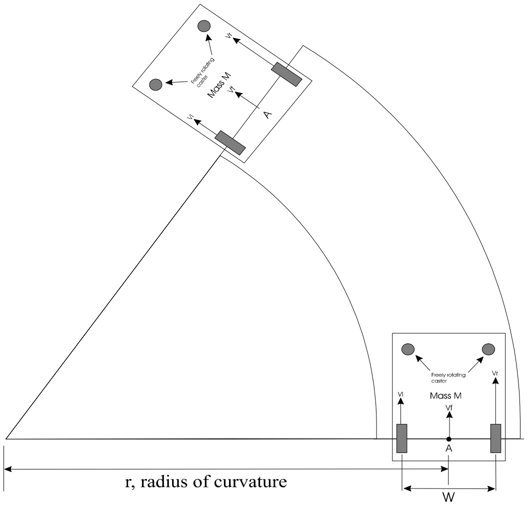

Figure 5--Vehicle Executing a typical turn

Using the definitions in Figure 5, it can be shown that Equation 1 describes the forward kinematics for the system. That is, for any desired forward velocity and rate of turn, we can compute the required wheel speeds, limited only by the physical capability of the wheel motors.

$$\begin{bmatrix} V_l \\ V_r \end{bmatrix} = \begin{bmatrix} 1 & -\frac{W}{2} \\ 1 & \frac{W}{2} \end{bmatrix} \begin{bmatrix} V_f \\ \dot{\theta} \end{bmatrix}$$

Equation 1

The inverse kinematics, obtained by rearranging Equation 1 is given in Equation 2.

$$\begin{bmatrix} V_f \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} \frac{1}{2} & \frac{1}{2} \\ -\frac{1}{W} & \frac{1}{W} \end{bmatrix} \begin{bmatrix} V_l \\ V_r \end{bmatrix}$$

Equation 2

Equation 2 is implemented in the microcontroller to convert speed and steer commands into right and left motor speeds. It can be shown that for any vehicle speed and steer commands, the vehicle can be made to travel in a circular arc. Driving straight forward represents the special case of an arc of infinite radius, and rotating with no forward motion is the special case of an arc with zero radius. This concept of driving in circular arcs will be used extensively in the Path Planner.

Vehicle Simulation with Matlab

A diagram of the simulation setup is shown below in Figure 6. There are three main components in this vehicle simulation. (1) The Control System, which provides control signals to the Vehicle

Dynamics, based on the vehicle's perceived deviation from the track. (2) The Vehicle Dynamics portion reacts to the control signals according to the vehicle's dynamic model to update the vehicle's position and heading in world coordinates. This vehicle state in turn feeds into the third part, the Scenery Animation Driver (3). This portion returns the coordinates of the left and right edges of the track from the driver's perspective at the given vehicle state. This information is the deviation from the track, which serves as the basis for the controller decisions.

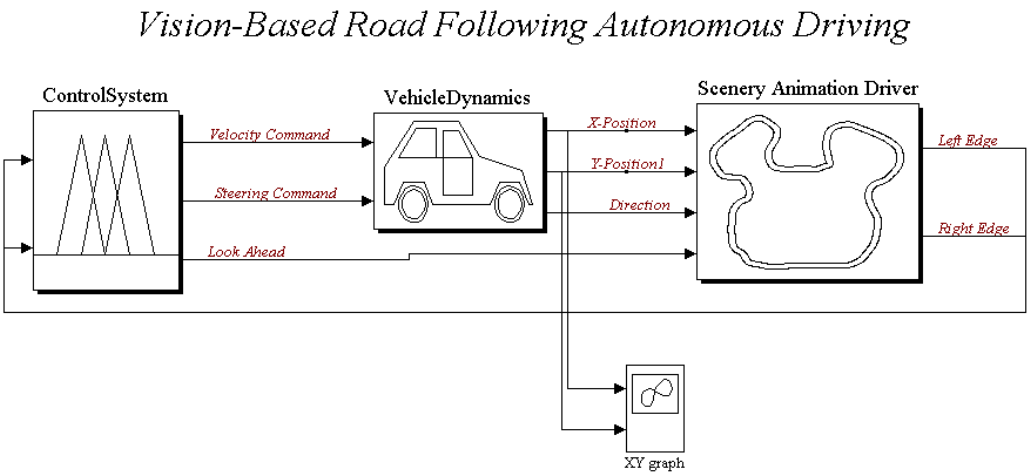

Figure 6—System Simulation Setup

Vehicle Dynamics

A block diagram for the vehicle dynamics is given below in Figure 7. The system is composed of three main blocks. (1) The forward kinematics block which translates the velocity and curvature commands into left and right wheel velocity commands. (2) The motor dynamics responds to the commanded velocity, producing the actual motor velocity. (3) The inverse kinematics block translates from the left and right wheel velocities into the actual vehicle velocity and heading. The remaining block relates the vehicle velocity and heading into world coordinates.

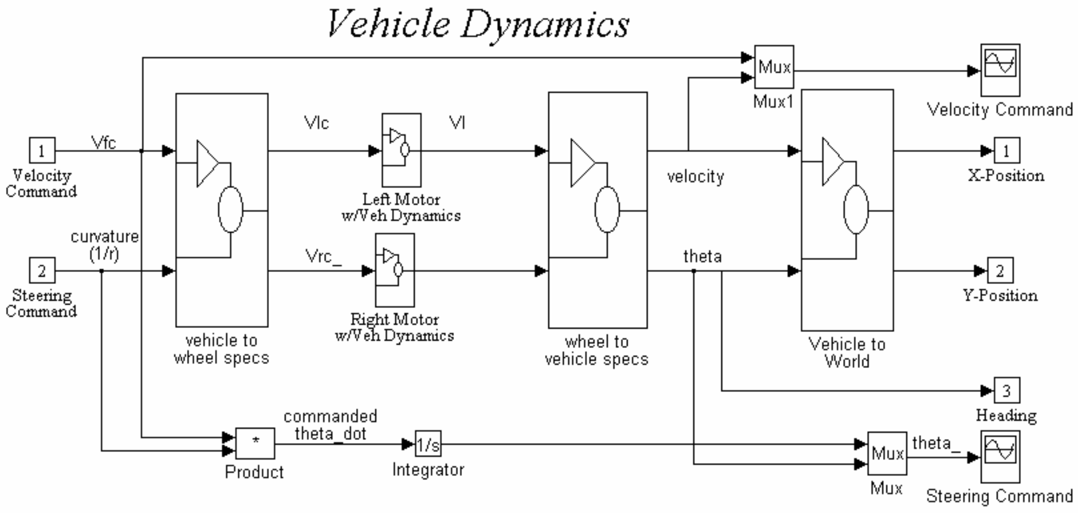

Figure 7—Vehicle Dynamics

Boundary Models

Left and Right boundaries are modeled by linearly interpolating the points that make up each boundary. The interpolation is only reliable if the data points from the sensors can be trusted as accurate. An example of the boundary models is shown in Figure 8. By modeling all relevant features on the track as boundaries, sensor data is more easily integrated together. As long as a given sensor can provide data about the track in terms of points, that sensor can be integrated into the system.

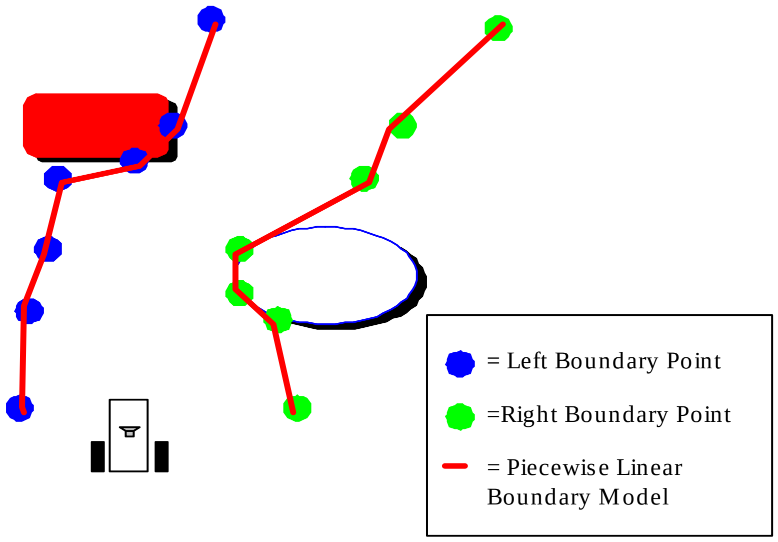

Figure 8—Left and Right Boundary Models

Path Planning

In the X--MAN system, all vehicle motions are modeled as circular arcs. Doing so allows us to plan vehicle movements in a manner consistent with the vehicle's natural motion (see Vehicle Kinematics). By combining this notion with that of boundary models, path planning becomes the task of *choosing the circular arc which moves the vehicle farthest along the track without hitting any boundaries*. The steps involved are outlined below.

1) Place vision data points onto a map (the Top View).
2) Add LRF data points onto the Top View (Figure 9).

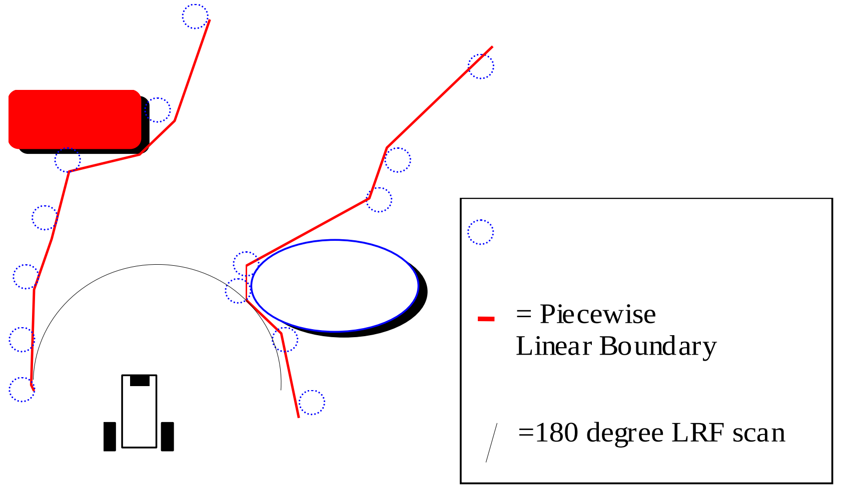

Figure 9—Vision and LRF Data Points, with Boundary Model

3) Evaluate a predefined set of potential arcs for the vehicle to drive along.*Interpolate*, in a piecewise linear fashion, the points that make up Left and Right boundaries (Figure 10)
4) Fit a circle to the center points, using a *Least Squares Estimate*. The circle must pass through the vehicle center, and be tangent to the vehicle's forward motion at the vehicle center (Figure).

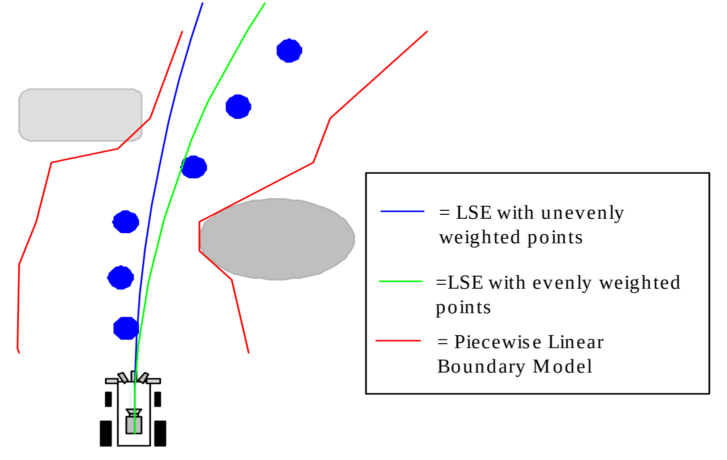

Figure 10—Path Planning: Circular Least Square Estimate

The steering control signal is the radius of the circle. Once the radius is determined, an appropriate speed to travel can be determined. Knowing these two facts, the required wheel speeds can be calculated and controlled (see Vehicle Kinematics).

Image Processing

The image processing associated with the camera images is one of the most important tasks for the main controller. One complete image has a resolution of 320x240, with 24 bits per pixel to represent the pixel's color, which computes to 225K. The challenge of the image processor is to examine the *information* in an image and extract the relevant track *features*.

In the X-MAN vision system, boundaries are distinguished by their color. All boundaries encountered on the track have a color in a well-defined set (white, red, yellow, etc.). Each examined pixel is determined to be either a boundary (in the set of boundary colors), or not a boundary (not in the set). The details of the decision making is given below.

Color Recognition

Each pixel in the camera image is represented as a combination of the primary colors red, green, and blue. That representation uses 3 bytes: one for red, one for green, and one for blue. Each byte represents the intensity of that color component. For example, if the 'red' byte contains a 0, then that pixel has no red component to it. If it contains a 255, then that pixel contains the maximum amount of red possible. Any color in the image can be described by an rgb triplet, a shorthand notation for expressing the color components. For example (r ,g ,b) = (0, 0, 0) is a black pixel, and (r, g, b) = (255, 255, 255) is a white pixel. All rgb triplets can be imagined to exist in a 3-dimensional color space.

Image Parsing

Color recognition is only one part of the image processing task. The other vital piece is image parsing, or the process of examining groups of pixels to extract features from the image. Whereas color recognition (in this system) applies to only one pixel at a time, image parsing applies to blocks of pixels in the image.

The primary function of the vision system is to provide boundary points to the Main Controller. Since a typical image contains so much data, it is important to process as little of it as is possible to extract features, in order to maintain a high controller update rate. The method adopted here is to examine *bands* of pixels, rather than the entire image.

The image parser divides the image into a number of bands. The image parsing is done within these bands. Starting in the center of the bottom band, a rectangle is defined within the band. The rectangle has the same height as the band, and a length Window Length. The color of each pixel within the rectangle is determined using the color recognition scheme detailed above. If the percentage of pixels with boundary colors is greater than a threshold value (the Minimum Window Percentage), then the position of the rectangle is noted. If the percentage of pixels with boundary colors is below the Minimum Window Percentage, then the rectangle is moved one pixel to the left, and the pixels within the rectangle reassessed. The rectangle moves to the left in this fashion until a boundary is encountered. Once a boundary is found, the rectangle returns to the center and slides to the right in search of the right-hand boundary.

Once the left and right boundaries are located in the first band, the midpoint of the two is calculated and searching in the next band starts at the calculated midpoint. This allows the parsing algorithm to remain between the left and right boundaries, even if they wander all over the image. Figure 11 demonstrates the parsing method.

A few remaining details must be explained. If the parser does not find a boundary, it assumes the boundary is where it was in the last frame. This allows the vehicle to drive along dashed lines, as it “remembers” where the lines were the last time they were seen. Also, the parsing starts at the bottom center of the image only in the first image of operation. Afterwards, it keeps track of the midpoint of the lane at the bottom of the image and starts parsing there in each image.

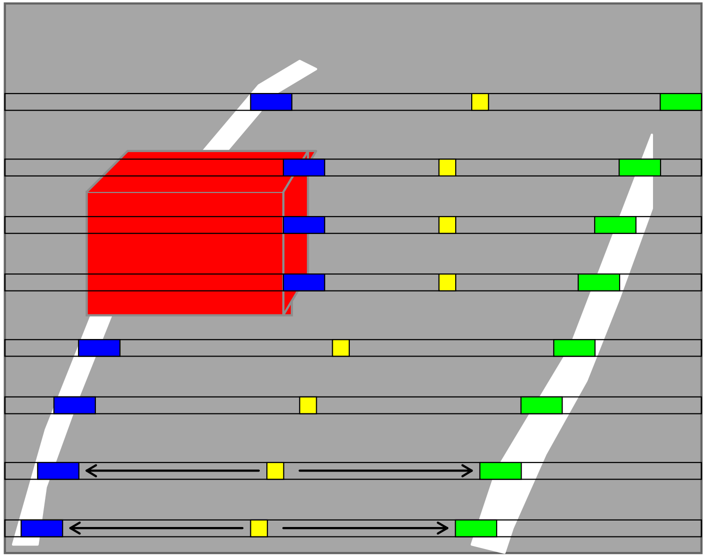

Figure 11

After an image is completely parsed, all the left and right boundary points found in the image are converted to their respective coordinates on the ground with respect to the vehicle. These coordinates are then passed to the Path Planner portion of the controller, which chooses an appropriate path to follow based on the vision data.

Experimental Analysis

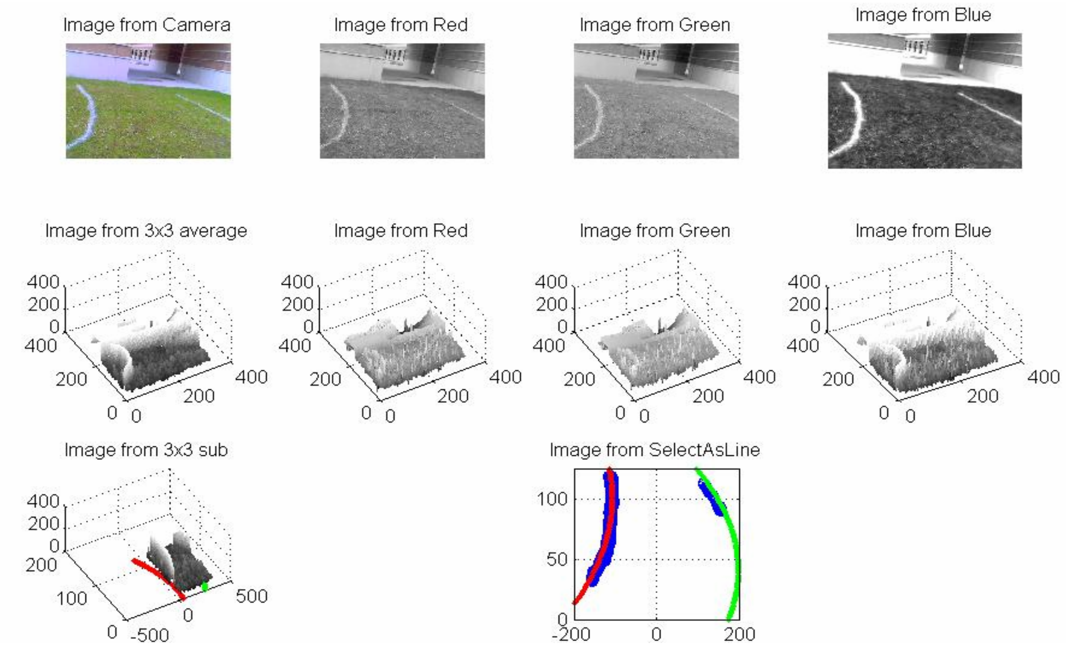

Figure 12.

The above figure shows the actual image of the lanes from the Windows based camera. This image is then displayed with respect the R, G and B components both in 2D as well as in 3D as above. From the forth figure ‘Image from Blue’, it can be understood that the image with just the blue components shows clear visibility of the lanes. This image is further processed with several image processing techniques to amplify the visibility of the white lanes. It is from this image, we determine the lanes as shown in the above in ‘Image from SelectAsLine’ with red and green lines using polynomial fit.

Once the lanes are successfully determined, the mid points are calculated between them. Based on the position of the midpoints path is determined and control signals are generated to maneuver the Robot. The following figure shows the midpoints between the lanes.

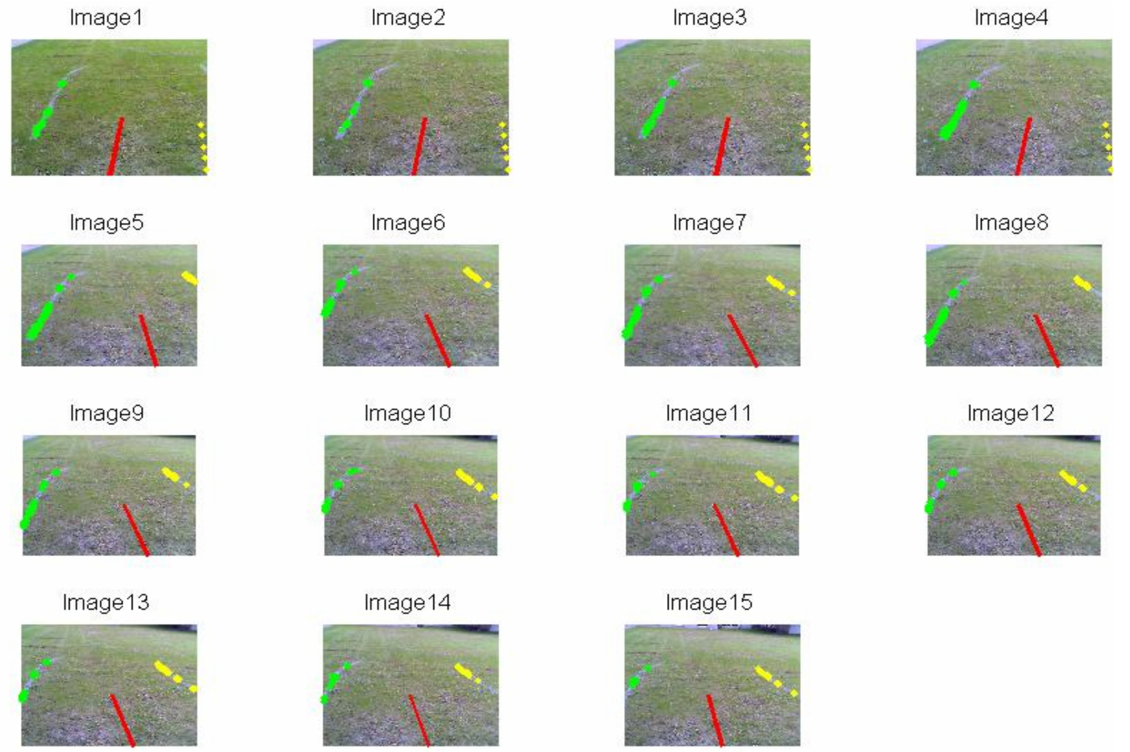

Figure 13.

The midpoint is represented with the red line and green and yellow represent the right and left lanes, which can be seen from the above Figure 13.

Design Process

The design process involved during the construction of X-MAN was at the same time formal and informal. The methods employed involve several distinct steps, including brainstorming, simulation, construction, and testing. All the steps together form a cycle. During the project, different aspects of the vehicle were always in different steps of the design process, but as time goes on, there is a general trend from the “Brainstorming” area of the flowchart to the “Testing/Evaluation” area. Our design process is shown in Figure 14.

An informal manner of communication was adopted. The small size of the team made it possible to eschew formal meetings in favor of daily contact among all the team members. This arrangement made it possible for all members to coordinate their actions with the others, resulting in no ambiguity about the status of any given topic.

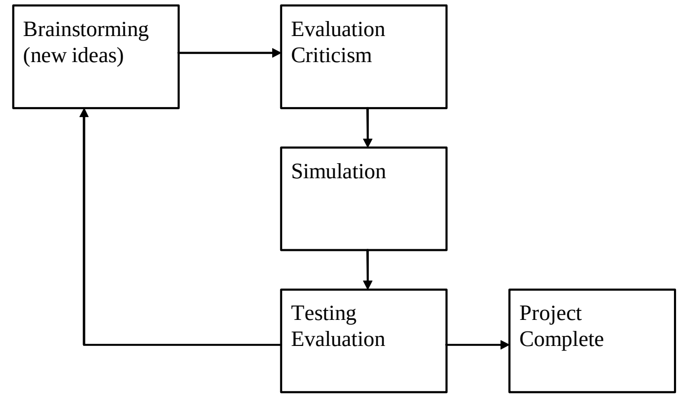

Figure 14—Design Process

Estimated Cost

X-MAN was designed on a very small budget. However, quality design and construction were still maintained through judicious and frugal assessment of requirements. Wherever possible, components available from other projects were borrowed or accepted as donations. The following table summarizes the financial aspects of constructing X--MAN.

System or Item	Cost (or value if donated)
Friendly Robotics RoboMower	$400
Laptop	$2000
Web Camera	$100
Speed Control & Related Hardware	$250
Sick LRF	$5000
Power Distribution System	$250
Equipment platform	$750
Service Cart	$250
Total	**$9000**

Conclusions

Designing and building X-MAN has been an intense experience for all team members. By providing an opportunity and motivation to apply concepts learned in the classroom, AUVS and Oakland University have allowed us a taste of designing real-world systems with real constraints. Although prototypical in construction, we believe the underlying concepts which bind X-MAN together possess far-reaching potential for more research and student involvement. Involvement with projects of this nature in a university setting provides a very rewarding way to learn about system modeling, programming, integration, and debugging. These features make the experience worth the effort for any student with will to participate.