UM-D WOLF

University of Michigan-Dearborn

Stefan Filipek, Richard Herrell, Jonathan Hyland, Edward Klacza,
Anthony Lucente, Sibu Varughese

Department of Electrical and Computer Engineering
University of Michigan-Dearborn

June 8-11th, 2007

Intelligent Systems Club
ISC
University of Michigan-Dearborn

1. Introduction

UM-D Wolf was designed to be a multi-functional vehicle that can operate in most terrains. This vehicle was designed and built from scratch by the Intelligent Systems Club, a student organization at the University of Michigan-Dearborn. The final objective was to create a safe and robust robotic platform for almost any application that can easily be incorporated into a wide array of electronics and sensors. UM-D Wolf design is geared for the 15th Annual IGVC competition, but is certainly not limited to this specific function. The platform is compatible with almost any new hardware, software, and variation in functionality. Since the concept of the architecture is simple and effective, the concept can be scaled to a smaller powered vehicle, a larger military vehicle, or anything in between.

2. Design Innovations

The most important innovation is UM-D Wolf's simple, yet versatile architecture. Since the vehicle is controlled by UDP packets, there are many devices that can operate it: a handheld 802.11 remote, a laptop or PDA with wireless capability, or even a computer hardwired into the network, located directly on the robot.

Secondly, the power of the robot is easily changeable and long lasting. The 24V system can be changed to parallel 12V system to allow for an extremely long battery life of over 12 hours without using a gas generator or charger. This estimate is for continuous use, so if the robot was not in continuous use, the power would last for days on end.

Furthermore, the motor controller is extremely efficient when it comes to power. It can be configured to ramp up speed on the motors to eliminate high start currents.

Without a ramping of the motor speed, startup currents can reach up to 100A. By changing the ramping speed, the max startup current is around 25A and only requires around 8A to keep running. Power distribution to the motors can also be controlled. If the motor on channel 1 runs a little slower than the motor on channel 2, the motor controller can be programmed to send a little more power to the first motor so that the motors on both channels will spin in sync.

The software itself is also innovative. The main program is written in such a way that if a critical system should fail, the robot will enter safe mode. For example, if the robot's E-stop is hit and the motor controller turned off, the code will know that the motor controller is not present. The robot will not operate again until the motor controller is online. Also, if a ½ second goes by and the robot has not received a UDP packet, the robot will go into safe mode and stop. This prevents the robot from running away in an autonomous and remote control situation. If the robot becomes out of range, or the transmitter loses power, then the robot will simply halt until it gets a command instead of carrying out its last command.

3. General System Design

A well designed base vehicle platform is a large factor contributing to winning any competition. The chassis of the robot should be sturdy and can support travel up to 5 mph, while providing the mobility to make fast and sharp turns. The power distribution system must provide power to motors and camera and still be easily accessible for debugging problems. The vehicle platform, shown in the photo below, successfully executed all the design goals.

-- Close up view of the vehicle --

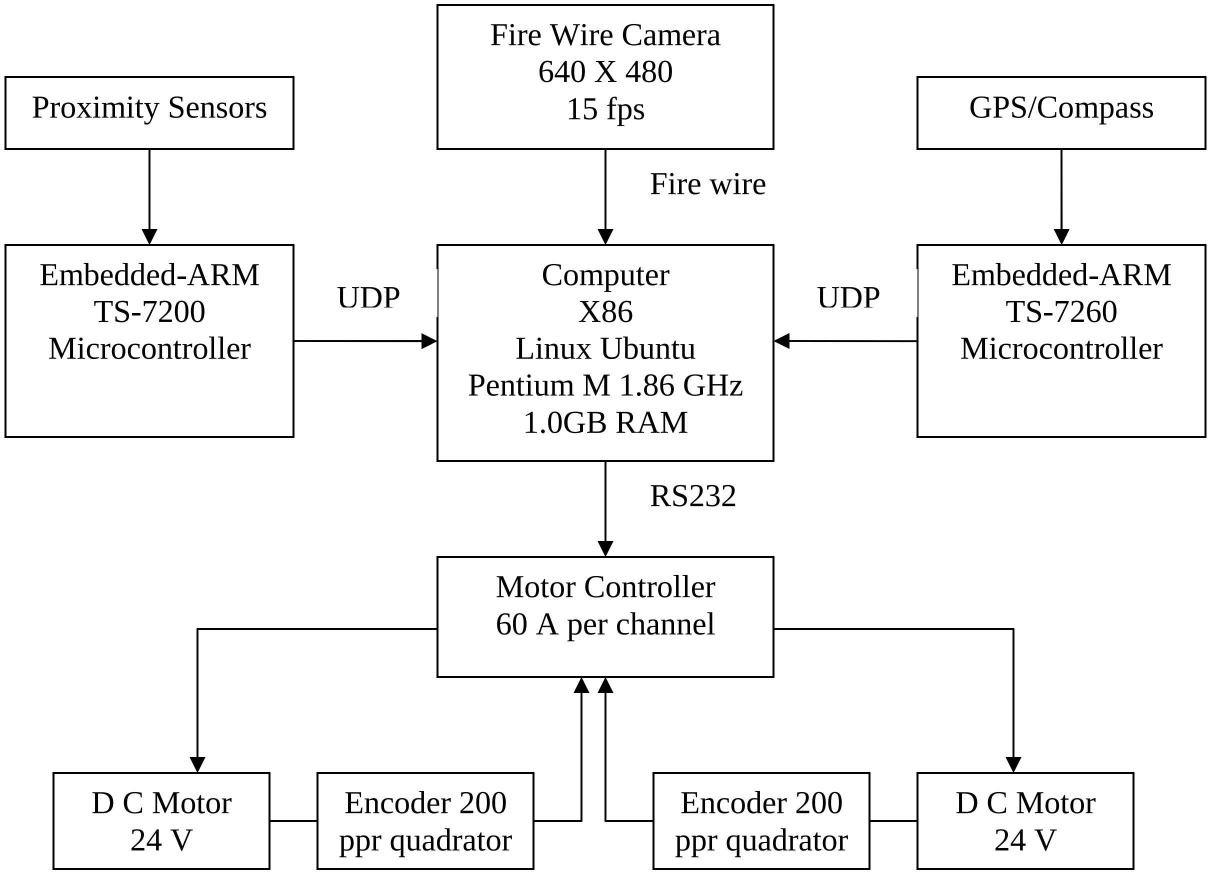
UM-D Wolf uses two differentially driven wheels in the rear for propulsion and two casters in the front for stability. The previous designs placed the casters in the back, causing the vehicle to tip when executing a fast stop. Another improvement to stability is a basin shaped interior that allows components to be mounted below the drive axle, lowering the center of gravity. UM-D Wolf is equipped with two 12 V deep cycle batteries, which provides continuous run time of at least 6 hours. The electrical components of the vehicle are placed at the bottom of the basin. This protects the highly sensitive parts, and at the same time is easily accessible for debugging, as it is out of the way but can be accessed by removing just one piece of plywood. The fully loaded Wolf weighs 190 pounds with a 3 x 2 ft. body and a 1.5 foot camera mount.

3.1 Team Organization

Each member of this design team was assigned both a primary and secondary task to complete on the vehicle. However, our team frequently worked together when we encountered serious problems. As a result, the entire team, rather than just a certain individual made all critical design decisions.

4. Hardware Design

The hardware of the robot consists of three sections; the drive, the control and the perception. The robot has a custom chassis, which was tailored to meet the IGVC requirements. The chassis is made of welded metal bars and sheet metal. The platform is strong enough to hold over 150 lbs and sturdy enough to make sharp turns at 5 miles an hour. The whole system is compact and easily maneuverable. Two electric wheel chair motors are used to drive the robot. The control part consists of a Roboteq AX2580 motor controller. It also has optical encoders to give precise information about the speed. The motor controller is connected to a laptop with a Pentium dual core processor, which does all the image processing. For perception, the laptop is connected to a Uni-Brain Fire-I digital camera and uses proximity sensor to detect obstacles. See system Architecture diagram below:



-- System Architecture --

4.1 Motor Controller

UM-D Wolf uses the Roboteq AX2850 motor controller, which is equipped with two channels to provide commands to the motors. Both channels of the motor controller are connected to each motor with an Anderson connector. The motor controller is mounted to the rear portion of the hull. The motor controller is setup to communicate through RS232. If a message is not received by the motor controller at least once per

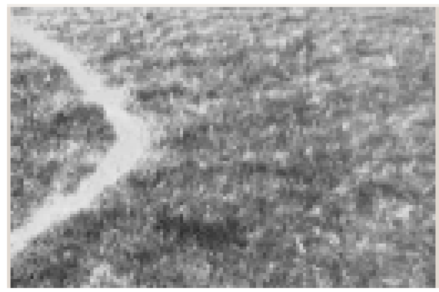
second, the watchdog timer in the motor controller will trigger the motor controller to safe itself. Although the RS232 communication portion of the controller can not be changed, there are several ways to communicate with it, either by PWM or a simple 0-5V signal. In its current configuration, the motor controllers communicate through RS232 with the main processor on the robot. Almost every other option on the controller is configurable: speed ramping, encoder usage, power distribution, etc.

5. Image Processing: Lane Detection

5.1 Binary Image Processing

The image from the camera is in 24 bit RGB (red, green and blue) format. Each pixel of the RGB image can be broken down to a set of three 8 bit values, with each value corresponding to the amount (intensity) of red, green and blue. In order to simplify the image processing, each pixel was reduced to a single value from the original set of values. By doing this, it shrinks the number of operations of image processing by two-thirds. This process of conversion is known as gray scale and the resulting image pixels will be stored as a single 8 bit number.

There are several methods to convert RGB image to gray scale. A simple method is to add up 1/3 of each color value for new gray scale intensity. This method was not used, as different colors appear to have different brightness to human eye. Another methods are the NTSC television standard, which uses a formula $Y = 0.299 R + 0.587 G + 0.114 B$ and the Intel's Image processing Library uses the formula $Y = 0.212671 R + 0.715160 G + 0.0721 B$. These two methods give a fairly accurate representation of the original image, especially in the image contrast and overall brightness. However, for extracting the lines, preserving the appearance is not as important, so a method that exaggerates the lines, rather than balancing the colors, was formulated. This method is called the mixed channel conversion from RGB to gray scale. In our method, 1/3 of green channel is subtracted from the blue channel. This method also reduces some noise from sources such as dead or brown grass. The figure below shows the gray scale image.



5.2 Thresholding

Thresholding helps the computer decide which pixels might correspond to lines painted on the grass. At the end of this process, the initial camera image will be converted to a binary representation where white pixels represent lines and black pixels represent free space. A popular approach is to apply a pixel intensity threshold. The simplest approach in performing an intensity threshold is to set an intensity level where brighter pixels are considered part of a line and darker pixels are free space.

In this project the threshold value was chosen to be 200. The figure below shows the threshold image of the course.

Threshold image

5.3 Morphology

Mathematical Morphology is a tool for extracting image components that are useful for representation and description. It is a set-theoretic method of image analysis providing a quantitative description of geometrical structures. Morphology can provide boundaries of objects, their skeletons, and their convex hulls. It is also useful for many pre- and post-processing techniques, especially in edge thinning and pruning.

In this project, Morphology is used multiple times. The first is the black and white morphology, where the image is eroded with custom structuring element. This is to distinguish the white line from the background. After that, binary image morphology is done. In this morphology, an opening followed by closing is done with thinning. The image below shows the result of morphology.

-- Image after performing morphology --

5.4 Hough Transform

The Hough transform is a technique which can be used to isolate features of a particular shape within an image. Because it requires that the desired features be specified in some parametric form, the *classical* Hough transform is most commonly used for the detection of regular curves such as lines, circles, ellipses, *etc.* A *generalized* Hough transform can be employed in applications where a simple analytic description of a feature(s) is not possible. Due to the computational complexity of the generalized

Hough algorithm, we restrict the main focus of this discussion to the classical Hough transform. Despite its domain restrictions, the classical Hough transform (hereafter referred to without the *classical* prefix) retains many applications; most manufactured parts (and many anatomical parts investigated in medical imagery) contain feature boundaries which can be described by regular curves. The main advantage of the Hough transform technique is that it is tolerant of gaps in feature boundary descriptions and is relatively unaffected by image noise.

5.5 Robot steering

After vision processing and line detection, the vehicle heading had to be determined. Initially a simple algorithm which finds the mid-point was used. While testing, however, there was a lot of external noise due to various problems with the wide angle lens. In order to correct that, the lens was removed and finding the mid-point of the line was no longer feasible, so a totally different algorithm was developed. According to this system, the line detected in the algorithm was recorded in red and the vehicle was commanded to move away from red. In order to do that the image was split in half. If the red line is seen on the left side of the image, the vehicle will move to the right, if the red line is seen on the right side of the image, the vehicle will move to the left. From testing, it was certain that this approach works. The figure below shows a screen capture of the red line in the image.

6. Obstacle Detection

UM-D Wolf uses a set of various proximity sensors to achieve the goal of obstacle detection. Each different sensor has its own job to perform in eliminating the robots blind spots and mapping where the obstacles are. Sensor data is read by an Embedded ARM board, by either an analog to digital converter or digital inputs. The first line of defense is the robots Ultra-Sonic Sensors, which are located on the front and sides of the robot. These five sensors are used to give the robot knowledge that it is coming up on an obstacle, within a detection range of 10 feet. Once an object is detected the robot will slow down to a pre-defined speed and use its next line of defense, the five beam IR sensors. The five beam IR sensors are used to locate exactly where the obstacles are in the robots path, within a detection range of 5 feet. If those two scenarios somehow fail, UM-D Wolf is equipped with eight IR proximity sensors that will provide an indication if something is within nine inches from the robot. The obstacle's distances are placed in a map and a control algorithm was written to stop the vehicle if the obstacles are too close, or take evasive action for obstacles that it senses in the distance.

MaxSonar®-EZ1

Sharp GP2D15 IR Sensor

5 Beam IR Sensor

7. Navigation

Navigation data was acquired from a Garmin 16a GPS. The GPS coordinates were combined with movement and velocity information from wheel encoders on the vehicle in a Kalman filter to improve accuracy. The heading of the vehicle was estimated from the difference between successive GPS readings and updated using the wheel encoder data, also in the Kalman filter. The position and direction from the Kalman filter were used as the starting point in the path planning algorithm which was used to determine the least cost path towards the next waypoint. While traveling towards a waypoint, if an obstacle was detected, the robot recorded the position of the obstacle on its map and planed a path avoiding that obstacle and any obstacles which it may have encountered previously. This way the system can steer around obstacles while continuing to move in the general direction of the next waypoint. Because the GPS is corrected by WAAS and not a higher accuracy differential correction service, the accuracy can often be worse than 1 meter. The robot compensates for the low accuracy position by looping back through the same area before traveling to the next waypoint.

8. Safety, Reliability and Durability

Our team wanted to ensure the safety, durability and reliability of the vehicle and as a result, the following methods are implemented in the design:

8.1 Manual emergency stop

The manual emergency stop consists of red push button to stop the vehicle immediately. Pushing the stop button cuts off the power to the motor and locks the wheels by turning on the electronic brakes, bringing the vehicle to an immediate stop.

8.2 Wireless emergency stop

A wireless remote keyless entry unit with a range of 100ft (30 meters) was modified to stop the vehicle remotely. When the remote button is pressed, power is cut to the motor, stopping the motor immediately.

Detailed Cost Analysis

Listed below is a breakdown of the cost:

Description	Cost		Quantity	Total
Chassis	$250.00		1	$250.00
Wheels	$40.00		2	$80.00
Casters	$20.00		2	$40.00
Laptop Computer	$1000.00		1	$1000.00
Embedded ARM Microcontrollers	$555.00		2	$1110.00
MaxSonar-EZ1-Ultra-Sonic Sensor	$125.75		5	$125.75
5-Beam IR Sensor	$52.00		3	$156.00
9” IR Proximity Sensor	$12.50		8	$100.00
Motor Controller	$700.00		1	$700.00
Fire Wire Camera	$173.45		1	$173.45
Mounting Supplies	$49.47		1	$49.47
Batteries	$159.95		2	$319.90
Painting/ supplies	$28.90		1	$28.90
			Product	$ 4,133.47

Acknowledgments

The team would like to thank Vimlesh Shukla for his help and support for the project.