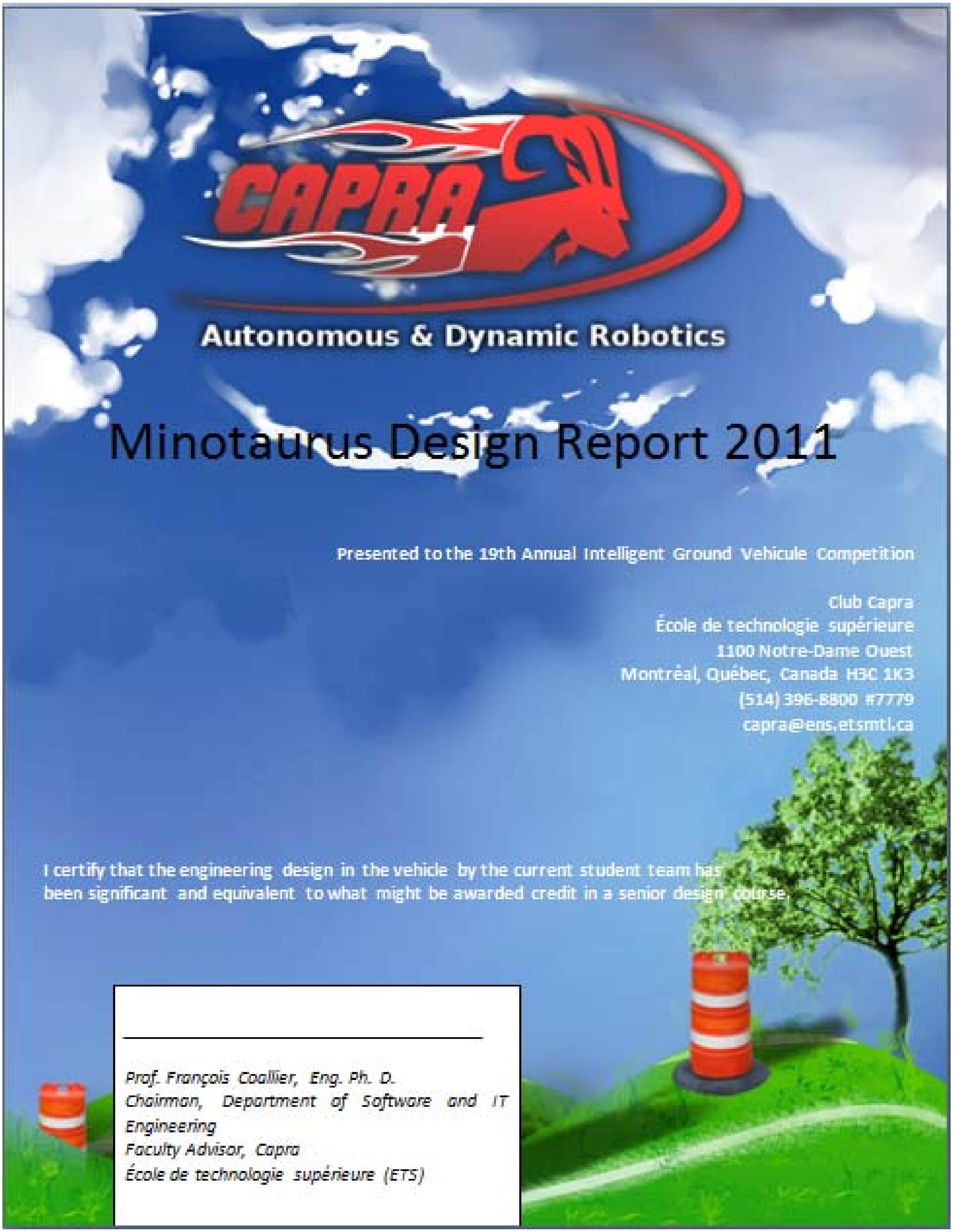


CAPRA

Autonomous & Dynamic Robotics

Minotaurus Design Report 2011

Presented to the 19th Annual Intelligent Ground Vehicle Competition

Club Capra
École de technologie supérieure
1100 Notre-Dame Ouest
Montréal, Québec, Canada H3C 1K3
(514) 396-8800 #7779
capra@ens.etsmtl.ca

I certify that the engineering design in the vehicle by the current student team has been significant and equivalent to what might be awarded credit in a senior design course.

____

Prof. François Coallier, Eng. Ph. D.
Chairman, Department of Software and IT Engineering
Faculty Advisor, Capra
École de technologie supérieure (ETS)

Table of content

Introduction

The club was founded in 1996 by a group of students passionate about the world of robotics. Capra is a student science club, which has as main goal the design and implementation of autonomous vehicle.

For almost 6 years now, our red robot, also known as RS3, have been serving the club with its autonomous skills. We have been using this robot to demonstrated and promoting stability with its hardware architecture. Once the last competition completed, we came to the conclusion that we need new technologies. It is with that thought in mind that Capra decided to build its fifth robot.

At the time of the creation of this document, the club is working on brand new robot: Minotaurus, that will be operational early in the month of June, time of the competition. This robot will be able to move toward cones, flags and white lines painted on the ground. This competition is worldwide and brings together the most prestigious universities in the field of robotics.

Team

With the mind of keeping it simple as possible, the project management has started for this new robot. Looking at the team's architecture, the team contains 20 actives students separate in 4 major departments who developed their new objectives and vision of this new robot. Here is the team's architecture and different teams:

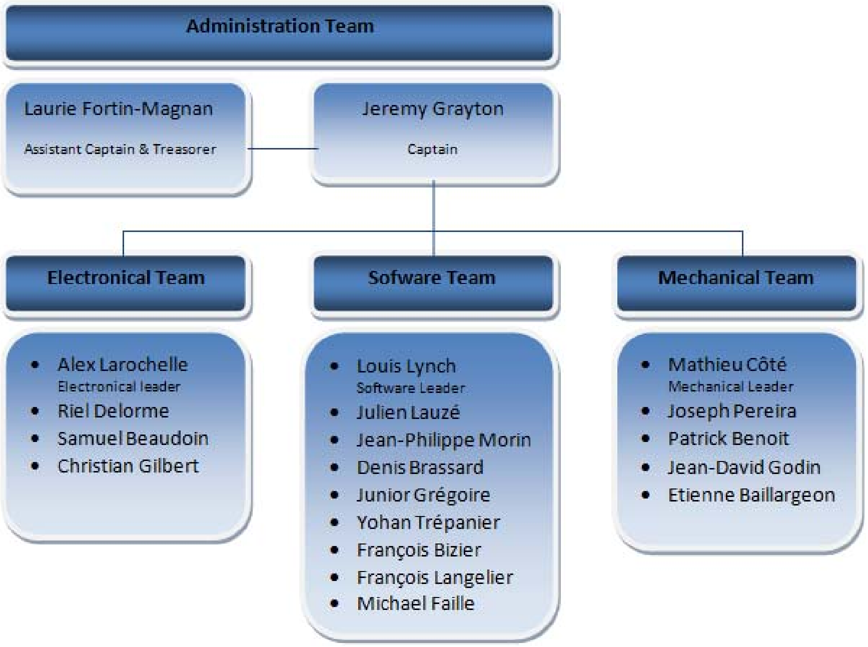
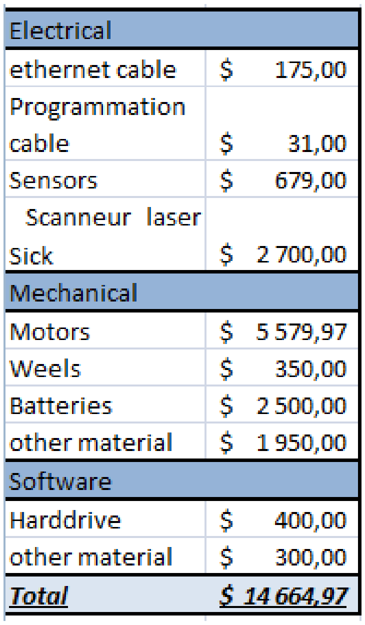


Figure 1- Team Capra architecture

Cost

Making a new robot need a lot of money because all new stuff has to be bought like the motors and batteries. This is an estimate cost for Minotaurus separate in different departments:



Electrical	
ethernet cable	$ 175,00
Programmation cable	$ 31,00
Sensors	$ 679,00
Scanneur laser Sick	$ 2 700,00
Mechanical	
Motors	$ 5 579,97
Weels	$ 350,00
Batteries	$ 2 500,00
other material	$ 1 950,00
Software	
Harddrive	$ 400,00
other material	$ 300,00
Total	***$ 14 664,97***

Figure 2- Cost of Minotaurus

Mechanical design

All along the development, the team has been using the criteria of keeping it simple for the design's guideline. By aiming a simpler robot then RS3 (older Capra's robot), we think the reability and the performance of Minotaurus will be increase beside RS3.

The fundamental concept of mechanical design is a four-wheel drive vehicle. In the past, Capra has try built an all wheel drive (AWD) propelling system, but it wasn't a success. We were using two wheelchair motor so the two right wheels was link together and as the two left one. The result was a robot with a difficulty to turn on the grass and using to much power. For Minautoraus, we are back with an AWD system but we're using four motors.

Structure of Minotaurus

The structure is in aluminum 6061-T6 square tube 1”x1”3/32”and completely welded. It contains two parts: the platform and the motor rack. The dimension of the robot is 40 inches long, 30 inches wide and 30 inches high. Furthermore, Minotaurus needed 24 inches

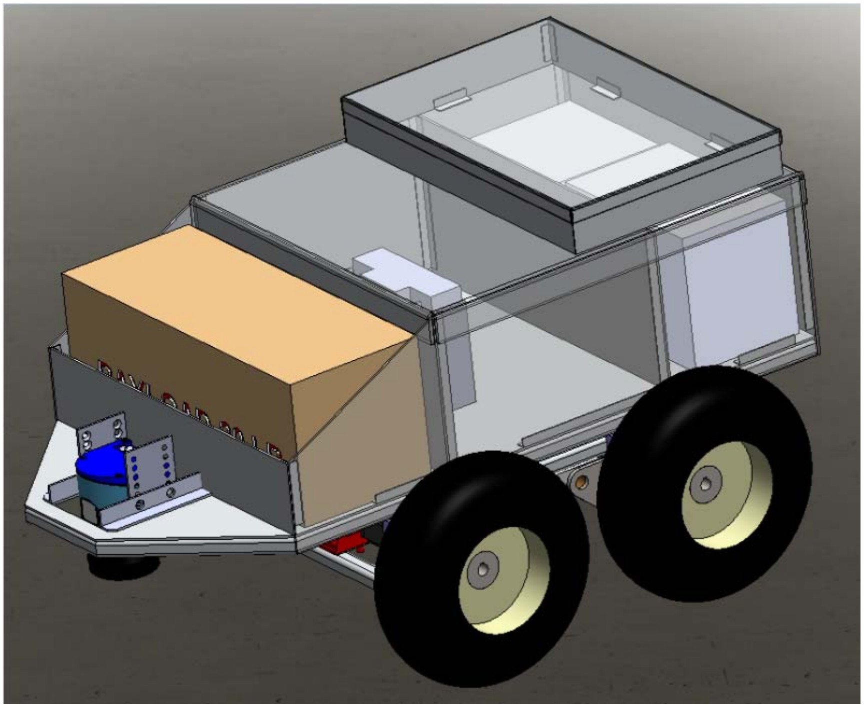


Figure 3- Minotaurus body

The platform is the upper part of the structure and also the frame. On the top, a ⅛ inches aluminum plate is used to hold all the electrical component. Under the frame, two stoppers are used to block the polyurethane bump-stop located on the motor rack.

The motor racks are mounted on the platform by pillow-block, and the brass bushing ensures the proper rotation of them. This system is a kind of swing arm suspension, there is a pivot, the bumps stop and the stopper to stop at the end of the course.

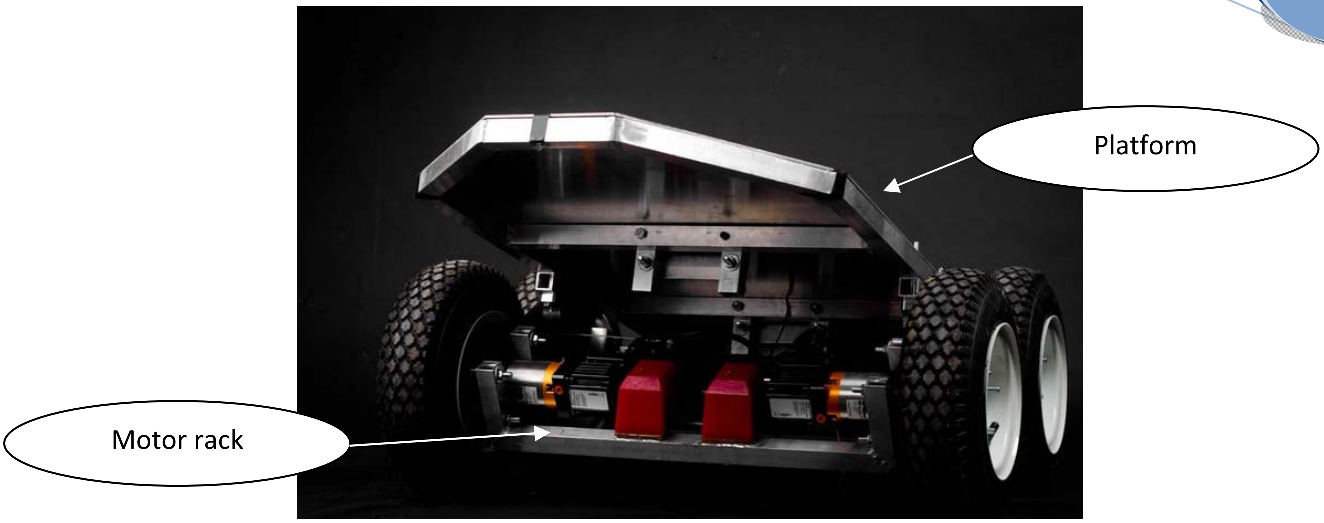


Figure 4- Frame structure

A light absorption system is put on Minotaurus, because last year the robot seems to lost control after the speed bump or when it was running at full speed. The absorption system will also avoid impacts on electrical components.

Drive train

The robot has four 12.5 inches lawn tractor pneumatic wheels. Each wheel is drive by an electric smart motor. They are ¼ Hp and we use four high precision gear box (ratio 20:1) to obtain the torque we needed to wheel. At low speed, they also give high accuracy. Because we are using four independent motors, we think that turning problem we had with older AWD Capra vehicle will be solve.

Moreover, this concept is to get the vehicle's center of rotation and the center on gravity at the same point. This means the four wheels will have the same traction.

As usual, a range finder has been placed in the front of the robot. With the specific form of the frame, the range finder can see in a widest range. In addition, the body is designed to be waterproof to protect all electrical components in case of rain. The case is in 0.063 inch 6061-T6 aluminium to have the best value for money with the perspective of minimum weight.

Electronics

The electric team has the task to link the software and the electrical part together. The goal is to create a safe system to control the robot based on the commands sent by the artificial intelligence.

Batteries

The new motors can be supplied from 24V to 48V but their torque curve peak is higher if supplied on 48V. To benefit from this higher torque, rechargeable 48V batteries are required, the batteries are lithium iron phosphate batteries because they're more stable, have a longer lifespan than any other lithium-ion battery and they are lighter than lead-acid batteries that we have on the other robot. The only security device we need with these batteries is a voltage comparator to make sure that they are not discharge below 44V to be sure that their lifespans are maximized.

To supply the other electronic devices via the power supply (dc/dc converter) we use two 12V cell series connected to give 24V by battery.

Power supply

The power supply convert 24 DC volt to a wide range of voltage: 5, 9, 12 and 19.1 to supply all our electric and electronic components like the inertial measurement unit, GPS, controller, camera and range finder. The new prototype power supply is smaller than Capra's previous model. It can also be used for recharging the laptop battery if it's empty and if the vehicle still needs to be running. The robot also has a monitoring integrated circuit that warns us with a buzzer when the batteries are low.

System signal processing

There are two parts in the electrical system, each one connected to a battery pack: the electronic section (computer, sensors and controller) and the motors section (motors and drive). This allows us to change the motor battery without having to shut down the computer and protect the computer from the power surge caused by the battery. The electronic section is powered by a 24v battery pack through a homemade power supply. This one is compact because it was made to fulfill our particular needs. A easy color code for the wiring is use to allows anyone to work on the robot with a minimal risk. The motor part is powered by a 48V 30Ah LiFePO4 battery pack.

The robot also has two safety devices. One is a simply emergency stop button that cuts power everywhere in the robot. The second one, is a remote emergency stop that does the same thing has the first one, but we don't need to be near the robot.

Every data is send to the laptop, by different communication protocol (USB, firewire, RS232, etc.). The A.I analyzes the data and takes the right decision. The camera send image to the laptop for the treatment. The laser scanner sends the distance between the robot and the obstacle. The motor send back the speed they are going. The A.I takes all those data and computes them to make the good decision. The A.I also sends the speed command to the motor and autonomous signal to the microcontroller. After that, the microcontroller opens the light or makes them flash depending on the command received.

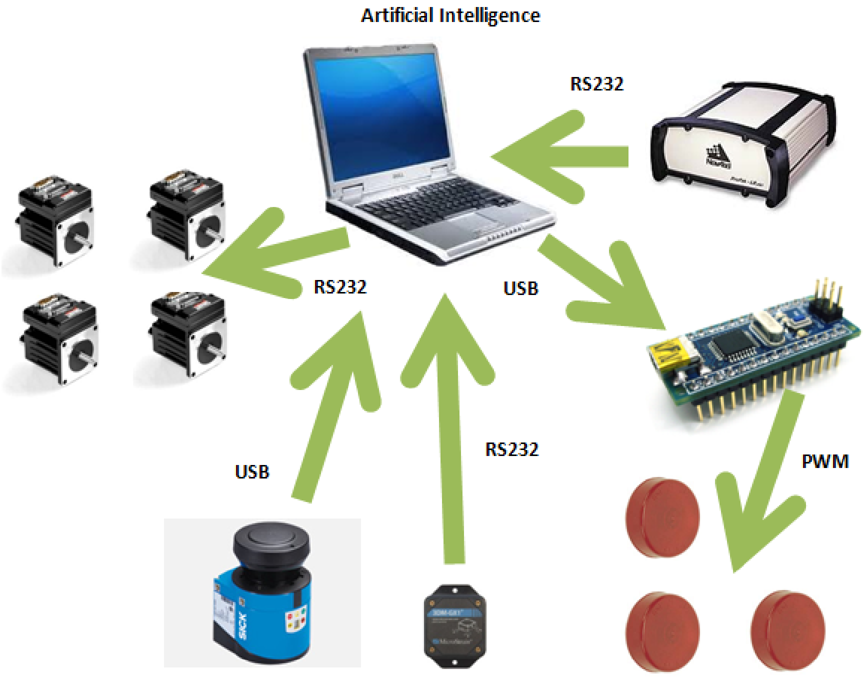


Figure 5- Communication between the sensors and the computer

Wiring and diagram

With the new robot, we try to avoid doing the same mistake than the previous years. Indeed, we now have an electrical wiring diagram to be sure that every member of Capra knows where every wire is from and where he's going. To achieve this goal, every wire is identifying with a number glued on it. Another mistake we've made is the "quality" of the wiring because every wire was mixed together. In Minotaurus, we will use spiral wrap to do a clean job.

USB Microcontroller

This year, we decided to buy an USB microcontroller to reduce the space we were using with the homemade version of our microcontroller. Due to the lack of time, it will only be use for the flashing light to indicate when Minotaurus is in autonomous mode. It will be use for more application by the next years.

Range Finder

The range finder has a scanning angle of 270 degrees and can communicate with ethernet. It's an upgrade versus the old one that we have on the other robot because it's smaller, less energy consuming and a faster communication rate.

Component management

During the conception of Minotaurus, the idea was to keep it simple as possible so the electrical team tried to minimize the components by using components that gives more for less. The best examples of these components are the smart motors that are auto servo-controlled and removed the necessity for a motor controller and a motor drive to control the motors' speed. This also removes unnecessary wire because of their integrated drive. We used the same philosophy with the USB microcontroller, USB hub and the range finder.

Software

Managing Versions and Dependencies: Maven

This year, a new system is use to help us maintain dependencies among projects the solution. We always implement new drivers or modules that have independent libraries. Managing these dependencies with the version control system is quite hard.

Maven is responsible for resolving dependencies from a description. That description of dependencies is maintained in the version control system. This allows a developer to be able work on a single project at a time without having to check out the dozens of project we have.

Flashing light and Controller

A flashing light is need for the competition so this one flashes when the robot is in autonomous mode and it keeps open when the robot is not in other mode. A LED controlled by a micro-controller is use to make this task. We have put code for flashing LED in the controller. The Robot communicates with controller through a USB cable to signal its current mode. When AI is in operation, it communicate at every 5 seconds it's awaken state and micro-controller verify every 5 seconds if the awaken signal has been transmitted. If controllers receive a signal from the robot, it makes flashed the LED. At the moment, it don't receive a signal, it keep the light opened.

Motors

The robot has 4 Animatics Smartmotor wheel motors, one in each of the 4 wheels. They are servo-motors, and allow us a very precise control of their rotation. They all have a built-in micro controller that automatically sends the encoder counts to the motor. We can communicate with these micro controllers through a RS-232 interface. The multiple motors are connected one with the other, on a daisy-chain. Each motor has an address, ranging from 1 to 4.

The objective is to create a service that will be responsible of controlling the motors. That service will replace the old one that was used with the old motors. That way, it will automatically be compatible with the rest of our currently existing architecture. It will allow us to keep other parts of the system untouched, while using the new motors to their full potential.

The motors have three modes of operation:

- Velocity: The motor will turn with a certain speed, no matter how hard it is. This is the mode that is currently used to make the motor move forward.
- Torque: The motor will apply a desired torque that will be maintained even if resistance is applied. The speed is there for variable.
- Position: The motor aims a position. It will rotate to that position with the desired velocity, and then stop. This mode is very useful when the robot has to turn.

The devices have the capabilities to let us know the state of each individual motor. We can then diagnose a problem, may it be electrical or mechanical. For example, having a motor stuck in a hole would allow us to get out of troubles using the three others without damaging this one. We can also be aware when we try to use the motors in an inappropriate way.

Rangefinder

Minotaurus use many devices to be able to apprehend its environment. One of these devices is the Rangefinder that is used to locate obstacles around the robot.

The rangefinder is a device that uses a laser to find obstacles. It's placed in front of the robot awareness to locate the obstacles located on the robot's path. From the rangefinder we receive a table of polar coordinates and communicates them to the AI to transforms the table in Cartesian coordinates. With this new Cartesian table our AI can safely navigate avoiding obstacles.

Some older model of Rangefinder use serial connection. This type of connection is getting kind of archaic. It's less flexible than USB or ethernet since you have to know which port is connected to the device and you have no way you can determine if it's really plugged or not.
It can be connected to only one computer at a time. Serial port can use different type of configuration to communicate. If we don't know this configuration, the port has to be scan to find the correct configuration. The new rangefinder device uses another communication mode: TCP/IP the same protocol as Internet. This provides many advantages. First, the rangefinder can be connect to a router or a hub. This allows many computers to communicate and receive data from it. We can monitor it remotely while our AI process can use it. Moreover, the only thing that is needs to connect to the device is its IP and port, like any other services available on the network computer. It's also fast and it's usable from any language that supports sockets, it facilitates the implementation of drivers.

Vision descriptors

The vision system is composed of a server and a client. The server is responsible to grab images from the camera, apply filters and then to broadcast it to all connected clients.

Last year we were transferring the whole image matrix to the client. The client could then use this image to avoid going in the direction of white spots. The problem with this method is that our AI (the client) was implemented using Java and Java is less efficient than C/C++ to deal with images. We wanted to make the heavy work on the server side but still take the decision on the client side.

Our approach has matured since then. We now transfer a string representation to the AI system to allege its treatment. We have come up with several kinds of descriptors that can be used for communication. First we have experiment with sending coordinates of with spots instead of sending the whole image. Then we came up with a better algorithm using Hough transformation to detect lines, and the information transferred is minimal and more helpful than a cloud of unrelated points.

With the older version we had to deform the image to correct the scale that was deformed by the inclination of the lens. This transformation was expensive in term of processing time. Now this transformation is simply mapping pixel coordinates to the real environment positions.

This said, we have a more powerful computer this year (quad core i7 with 4GB of RAM). The vision execution time is optimized by eliminating unnecessary calculation. That relocated part of the treatment to a more efficient language (from Java to C/C++) for that particular purpose. We also have diminished the amount of data transferred between the vision server and its clients. All these modifications lead to a more reactive robot.

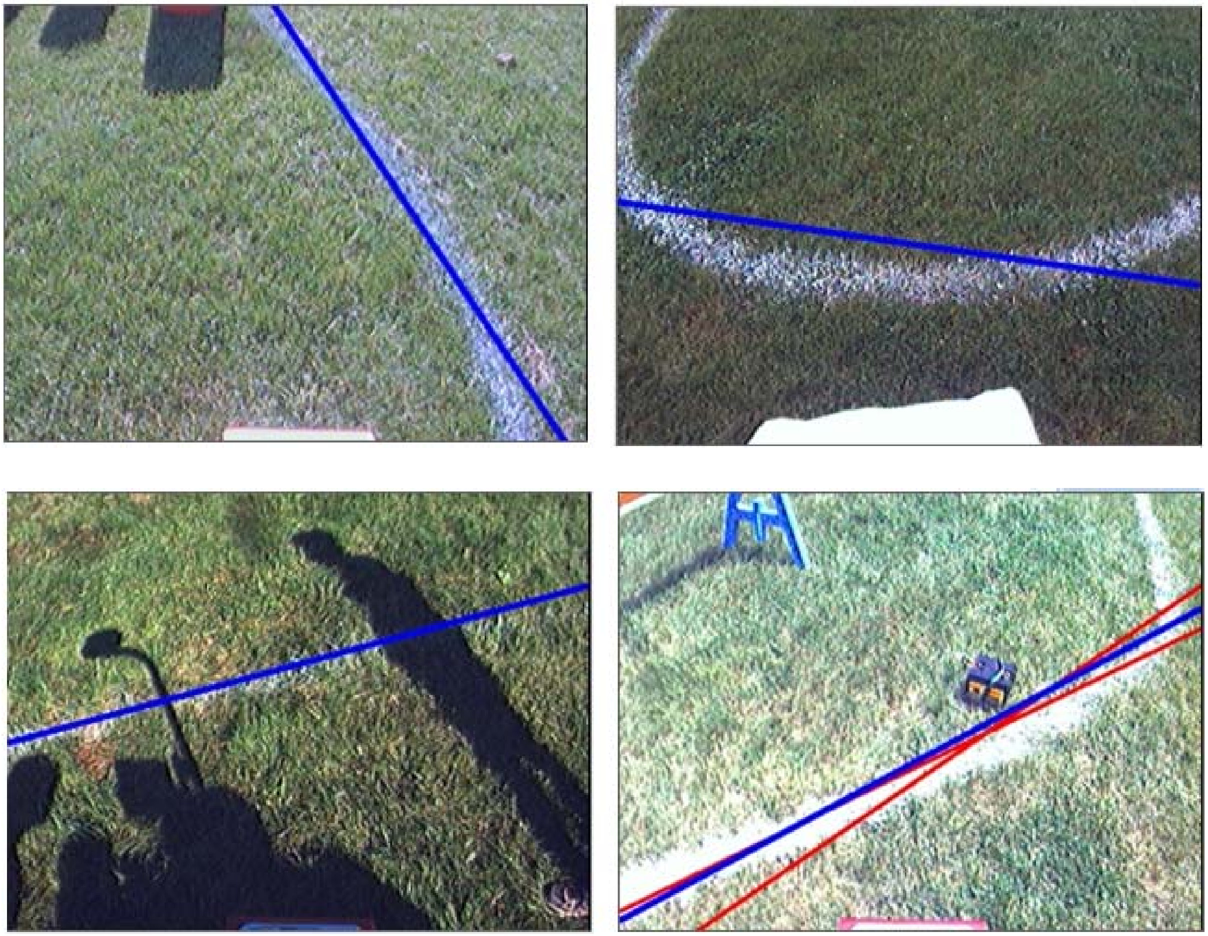
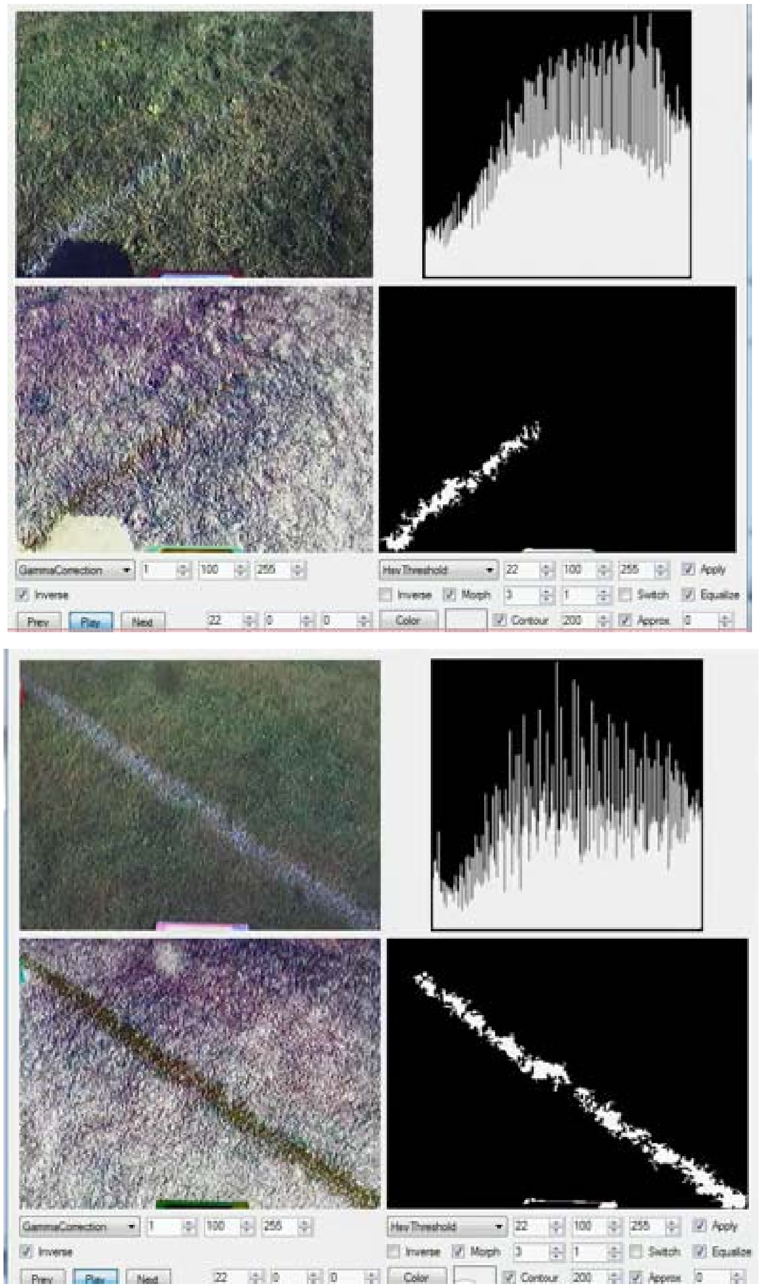


Figure 6- Orientation of the line approximate by the blue color

Visual Detection of Colors

Instead of fixing a threshold for color detection, a dynamic threshold is computed based on the histogram. This method tries to identify peaks of interests to find segmentation in the hue repartition of pixels. This technique is reliable since it can adapt to the ambient luminosity most of the time. This method is really good at detecting whites' shapes so it's used for lines detection.

Also to ease the detection of colors a program able to read from specific location of the vision feed has been implemented. This program is needed to see the differences between the grass and the green flag. By putting a green flag sample at the tip node of the robot better results are obtained.



GammaCorrection 1 100 255
☑ Inverse
Prev Play Next 22 0 0
HsvThreshold 22 100 255 ☑ Apply
☐ Inverse ☑ Morph 3 1 ☐ Switch ☑ Equalize
Color ☑ Contour 200 ☑ Approx 0

GammaCorrection 1 100 255
☑ Inverse
Prev Play Next 22 0 0
HsvThreshold 22 100 255 ☑ Apply
☐ Inverse ☑ Morph 3 1 ☐ Switch ☑ Equalize
Color ☑ Contour 200 ☑ Approx 0

Figure 7- Color detection processing

Taking decisions

For both the navigation and the autonomous challenge the rangefinder is the primary source of orientation. This device is fast enough to give us an accurate representation of the environment of the robot every inches. This is the only thing Navigation and autonomous have in common.

For the navigation challenge we use the compass and GPS in combination to orient the robot to next waypoint on the list. Using this technique it's always possible to correct the orientation of the robot to reach the goal.

For this competition the rangefinder is used since detecting obstacle with vision is not fast enough to allow us to navigate at the speed of navigation (2 meters by seconds) this would result in one frame every 20 centimeters. The vision algorithm still runs in case there would be a speed bump, in this case the robot would be slowing down to diminish the impact. Trying to change orientation at 2 meter by second using vision would be too risky so it's preferable to slow down and let the range finder decide for the high speed orientation.

On the autonomous field it's different. First the orientation and position of the robot is the least important part. We concentrate the resources on vision detection. Because the field of obstacle is denser than on the navigation field, we reduce the speed to 1 meter by second. At the speed it's possible to have a vision cycle executed every 10 centimeters.

The robot progress in this environment using heuristic based on obstacle detecting from the rangefinder and line detection from the vision server. Basically every ten seconds the robot receives new information about the orientation of the line in its vision. Using this information it's then possible to follow the lines and avoid obstacles even if lines are dashed, the algorithm will still remember it should orient itself the same way lines do.

JAUS

For the JAUS Challenge in the IGVC competition, we decided to start something from scratch since the last few years, we were not able to come out with something fully completed. Since, other student clubs at our school are using JAUS to communicate with their autonomous vehicle (a submarine and a drone), we worked to combine our efforts in the implementation of the Mobility JAUS Service Set and the JAUS Core Service Set. Sharing our comprehension with others and testing interoperability with them allowed us to achieve a more robust solution. With the implementation of those services, we developed a module that interprets the messages and transforms them so the robot can understand and respond.

To ensure that the implemented module was answering correctly to the JAUS commands received, we developed a tool test JAUS message by specifying appropriate values for the parameters. With our simulation tool, we can test the behavior of the interpreting module.

Conclusion

Team Capra is proud to participate in this multidisciplinary project and all members always try to surpass themselves. The students work for at least 500 hours on this new robot but that number is just a approximation. It will be the first competition for Minotaurus and all the team expect to have great results. We work hard this year to build a new robot and to keep it simple as possible in the conception. With new ideas and better applications in every department, Minotaurus will be the best robot Capra ever builds.