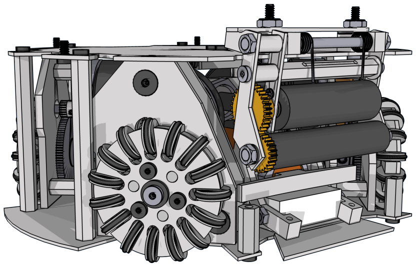
# RoboJackets 2008 Team Description Paper

Andrew Bardagjy *, Roman Shtylman, Stefan Posey, Jason Kulpe, Phillip Marks, Ben Johnson, John McConville, Stuart Donnan, Yasman Nemat, and Jake Nieman
* primary contact: andyb@gatech.edu

RoboJackets - www.robojackets.org
Georgia Institute of Technology - Atlanta, Georgia, USA.



**Abstract.** The 2008 Georgia Tech RoboJackets RoboCup team is using the techniques and lessons learned from our first year (2007) to improve and update all of its subsystems. To successfully compete, many systems were completely redesigned. This paper describes the development process and offers some insight into the decisions and trade offs which were made.

## 1 System and Team Overview

The robotic system is composed of three main subsystems: electrical, mechanical, and software. Each component has a small team of individuals working under a leader; the system leaders and the team leader communicate their findings and progress to one another to ensure a successful system. Each team sets its own goals and priorities with the team leader overseeing the entire process. As the team grows, this hierarchy has allowed the team to react accordingly. All team members are encouraged to participate at their own pace. Additionally, some effort is made to recruit and train new members to ensure the team's future.

## 2 Software

The software system consists of separate processes that communicate over UDP. Each component can start and stop transparently to the other processes. This makes it easy to quickly bring down, modify, and restart one part of the system without affecting unchanged components. Each particular packet type that needs to be sent is defined by a specific port number, and a different range of ports for each team. This means that the receiving process only needs to listen on the port of interest and know which packet type to expect.

Software is divided into 5 major components: vision, communication, motion, high level control, and simulation. The vision, communication, and simulation components are shared by both teams, while motion, and high level control are unique to the team they operate.

### 2.1 Vision

Two cameras are positioned over the field to provide sensing for ball position and robot position and orientation. The cameras are connected to a computer via gigabit Ethernet. Data from each camera is processed by a separate thread and sent to the rest of the system via multicast UDP packets. The vision process for each camera consists of six major parts: frame acquisition,color segmentation, spanning and grouping, distortion and coordinate transformation, identification and tracking.

Frame acquisition involves receiving a Bayer pattern image from the camera and converting it to a 24 bit-per-pixel RGB image. Frames can also be read from image or video files for testing. The frame rate of the camera controls the speed of the vision pipeline.

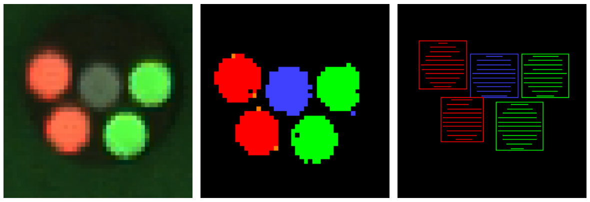


**Fig. 1.** Raw camera image, color segmented image, and spans and groups.

Color segmentation involves classifying each pixel in the image as belonging to zero or more of six color bins. The 24-bit RGB value is quantized to 12 bits. The color bins assigned to this quantized value are retrieved from a look-up table. A segmented image is generated for display and a list of pixel locations for each color bin is generated for further processing.

Spanning is the process of converting horizontal runs of pixels of the same color to a single structure. Gaps of limited length may be included in a span to reduce the effects of noise. Spanning is performed independently for each color bin. After spans are identified, overlapping spans on adjacent rows are combined into groups. Spans and groups are filtered based on criteria such as span length, group dimensions, and aspect ratio. Since an RGB pixel value can be assigned to multiple bins, there may be many pixels of a particular color which are noise and which will be ignored as they would form spans shorter than the minimum span length.

The center of each group is converted from image coordinates (in pixels) to world coordinates (in meters). A lens distortion model is applied to correct for radial distortion in the lens[1]. The undistorted coordinates are then linearly transformed into world coordinates.

At this point, each orange, blue, and yellow group is potentially an object in the game. Yellow and blue groups are possible robot centers. Robot identity and orientation are determined by the number and position of groups of different colors which are close enough to be on top of the robot lid. After potential robots and balls are identified, each is classified as either a new object, an object that is already being tracked, or a spurious group which will be discarded. The result of tracking objects is a list of up to five robots for each team and one ball. This object data is sent via a multicast UDP packet to the rest of the system.

#### Configuration

Most of the vision system is configured graphically. Color segmentation is configured by selecting pixels in the camera image whose RGB values are added to or removed from a particular color bin. The color segmented image is displayed with only one color bin for each pixel, although multiple bins may actually be assigned. The set of visible colors is user selectable.

The lens distortion parameters and coordinate transformation are also configured graphically. The auto-calibration method in Vass and Perlaki[1] is used to estimate lens distortion from points on known straight lines in the image. To correct for parallax, a separate coordinate transformation matrix is used for the tops of the robots and for the ball. Each of these matrices is calculated from three reference points manually selected in the image space with known world coordinates.

### 2.2 Communication

The communication process is responsible sending radio commands from the host to the radio base station and out to the robots. It handles all radio specific tasks as well as controlling the robots by mapping software IDs to physical robots. Communication accepts packets that specify the speeds of motors, roller state, and kicking power.

### 2.3 Motion

The motion process is responsible for controlling the motion and basic capabilities of the robots. Higher level components specify the intended function or location of the robots and allow the motion system to handle the detailed execution.

#### Behaviors and Constraints

Motion takes two parameter types: behaviors and constraints. Constraints limit the motion of a robot to particular locations and geometry on the field. Point constraints would make the robot stay at a certain point while line and segment constraints would limit the motion to the specified lines or segments. There are also constraints on angles for facing the ball, a particular robot, or a fixed angle. All of the constraint types are separated for motion control reasons. Specifying a constraint on the ball allows the motion to do better ball tracking when ball velocity is known. This allows for better tracking compared to setting the target angle from higher level control.

Behaviors are used as both modifications and additions to the constraints. Behaviors include such things as marking another player or handling the ball. When a behavior is active it changes the way certain constraints are handled in order to meet the specifics of the behavior. A simple example is retrieving a loose ball, where moving to an exact position is not necessary but speed and ability to block the other team is.

#### Path Planning

A novel path planner has been developed for this year's system. The algorithm examines all possible obstacles from the perspective of both the controlled robot and the intended destination. Obstacles are defined to be all other robots as well as zones to avoid the ball during kickoff and game stoppage.

*Algorithm:* Lines are drawn from the controlled robot to avoid all of the obstacles. Lines are then drawn from the destination to also avoid all of the obstacles. The intersection of any line in the first set with any line in the second set creates a two segment path. All of these paths are then tested for collisions with other obstacles and any path found to cause a collision is removed. This leaves only paths that do not cause a collision. Among those, the optimal path is chosen and the robot begins to traverse it. New path planning occurs when new vision information is available.

### 2.4 High Level Control

The current high level control has the ability to assign roles to particular robots. Roles can be assigned based on how well a particular robot is suited to the task. Selection criteria include both hardware capabilities as well as game state and pose information.

### 2.5 Simulation

A physics simulator for testing control and planning algorithms was developed for this year's team. The simulator replaces the vision and communications components such that no other changes must be made to other systems to use the simulator. The other components see the simulator just as they do the real systems.

## 3 Mechanical

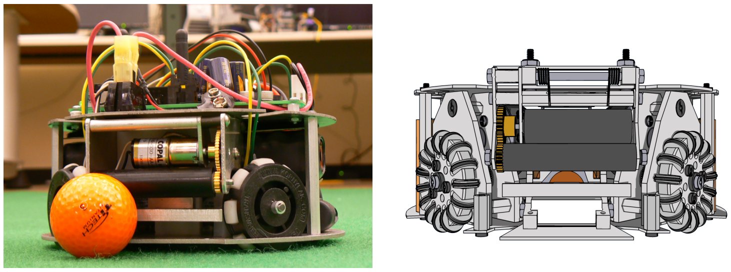


**Fig. 2.** 2007 Robot (left) vs. 2008 Robot CAD (right)

### 3.1 Testing

Testing has taken a central role in the development of this season's robots. Testing metrics have been developed for every mechanical subsystem including the dribbler, kicker, and chipper.

Dribbler testing focuses on evaluating the three dribbler control conditions as defined in Ruiz[2]. To simulate an interception, an apparatus has been designed which kicks the the ball with a pneumatic cylinder towards a stationary dribbler assembly. Designs are compared based on the maximum speed at which they can successfully control the ball and complete a pass. To test ball retention and holding power, the playing field is elevated at an angle relative to the ground. The dribbler torque can be estimated according to the maximum angle at which the dribbler can retain the ball. These tests allow comparisons among different dribbler roller materials and geometries. During the testing, many materials including neoprene heat shrink, silicon rubber, and santoprene were evaluated. Roller geometries include cylinders and various tapers on the dribbler bar.

Testing routines for the chipper and kicker are more straightforward. This involves shooting the ball at a target to determine accuracy, speed, efficiency, and maximum distance. During the testing different kicker and chipper geometries were evaluated to determine which would provide the highest accuracy and energy transfer.

### 3.2 Omniwheel

In the previous year, the robots were fitted with commercial-off-the-shelf plastic omniwheels.These wheels exhibited many problems including; poor power transfer, low traction, high orthogonal rolling resistance, rapid wear and deformation, and high roller bump height.

To alleviate these issues, custom omniwheels were developed. The omniwheels are composed of two plates of aircraft grade aluminum. Fifteen rollers fitted with neoprene o-rings allow for smooth motion and protection of the carpeted field or other running surfaces. These wheels offer substantial performance and reliability improvements over the previous year.

### 3.3 Drive Module

To further improve the driving performance and increase the internal volume available for other systems the 2008 robots use brushless motors and custom gearboxes. The brushless motors have a no load speed of 4370 rpm and a stall torque of 255 mNm. The 2007 motors had 10200 rpm and 27.1 mNm respectively. This allows for a smaller gear ratio and a narrower gearbox. The gear box consists of two spur gears with a 5:1 reduction. Carpet fiber build up was a major concern; therefore, the parts were designed to be easily replaced and cleaned. The drive modules are completely enclosed by the shell bringing the overall diameter to 179mm. This protects against ball and other robot impacts.

### 3.4 Dribbler

The dribbler is the assembly which controls the ball. It is designed to receive a ball kicked at 10 m/s. The dribbler has an aluminum bar on a shaft that is powered by a brushless motor with a 1:1.4 gear ratio between the motor and the roller. In 2007, the robot would lose control of the ball when moving backwards at 1 m/s. This year's dribbler was designed to allow the robot to moving backwards at up to 3 m/s. The final size and material of the dribbler roller will depend on further testing, however, initial experimentation with neoprene heat shrink has proven successful. The 2008 dribbler system pivots against torsion springs allowing the dribbler to "give" when catching the ball at high speeds. The springs along with a softer dribbler material should reduce the chances of the ball ricocheting after impact with the roller by absorbing some of the ball's kinetic energy. The dribbler was designed to have a ball coverage of no more than 19%.

### 3.5 Kicker

The robot's kicker is the primary method of both scoring and passing. A solenoid is mounted inside the robot and a large amount of current is discharged into it from a capacitor bank. Because the new robots use brushless motors, which provide more space in the robot versus the 2007 brushed motors, the kicker solenoid is mounted in line with the center of the ball. This limits the mechanical losses in the system, which severely hindered kicking ability in 2007.

### 3.6 Chipper

The chipper allows the robots to pass the ball by kicking it into the air. It shoots the ball over opponents and adds greater flexibility to planning algorithms while reducing chances of interception. Due to complexity issues the chipper will not be incorperated into this year's system.

## 4 Electrical

The electronics system is broken into two very broad subsystems; the controller and the kicker circuitry. The controller subsystem contains the radio link to the computers on the sidelines, and all motor control functionality. The kicker subsystem drives the kicker and chipper solenoids. To accommodate brushless drive and dribbler motors, the controller subsystem was entirely redesigned for this season. While the requirement to drive a chipper was added to the kicker subsystem, the charger and other switching mechanisms have remained largely the same.

### 4.1 Controller

The primary functionality of the controller subsystem is to translate motor speeds sent by the computer on the sidelines to actual motor values. Though the controller is currently ”dumb” we hope in the future to enable more intelligence on the robot.

From the requirements of the drive and dribbler motors, the requirements of the electrical system can be derived. Each motor has three phases connected in a wye configuration and three hall effect sensors to establish rotor position. To drive a brushless motor the rotor position is used to determine which coils should be energized. A coil is energized with a half bridge. Each half bridge is composed of an N and P channel FET driven by a Microchip FET driver (TC4428). There are three half bridges per motor (one for each coil) for a total of 15 half bridges per robot. With miscellaneous passives, the motor driver circuitry composes about 150 components on each board. The half bridges are driven by a Xilinx 100K gate Spartan 3E (XC3S100E) FPGA. The FPGA is memory mapped to a NXP ARM7 (LPC2103) which handles local feedback control.

The robot communicates to the server via Lynx HP3 wireless modules. They are routed through the FPGA to the ARM to allow for future work in hardware accelerated wireless protocol research. Due to poor wireless performance in the previous year, and severe space constraints in the current design, standard monopole antennas were not used. Instead, a balanced dipole "halo" antenna will be utilized. The antenna is ideally suited for the challenge as it is very low profile and omnidirectional [3].

On a typical design, power supplies and their distribution are normally considered a trivial implementation task. In contrast, this design has five different power rails running on the board; 1V2, 1V8, 2V5, 3V3, and VBATT (12V). Though it was determined that linear regulators could be used for all supplies other then the 3V3 supply, the many power rails still present a considerable routing challenge. Despite those challenges, the board is only two layers which affords much quicker and cheaper manufacturing then other processes.

### 4.2 Kicker

The kicker circuitry charges a bank of capacitors which are then discharged into the solenoids (both the kicker and the chipper) to kick the ball. The kicker uses a Linear Technologies LT3750 flyback controller to convert the battery voltage (12V) to approximately 300 volts which charges a two millifarad capacitor bank. The capacitors are discharged into the solenoids by means of an Insulated Gate Bipolar Transistor (IGBT). Ball speeds approaching legal limit have been achieved. In comparison to other designs, this design is quite compact, requiring only about five square inches of board area.

## 5 Conclusion

After many months of development, though the system is more advanced then the previous year, much progress has yet to be made before the competition in July. Nevertheless, we are confident in our ability to produce a competitive system and are excited about play on the international level.

## References

1. Vass, G., Perlaki, T.: Applying and Removing Lens Distortion in Post Production (2003)
2. Ruiz, M., Weitzenfeld, A.: Soccer Dribbler Design for the Eagle Knights RoboCup Small Size Robot. Robotics Symposium, 2006. LARS'06. IEEE 3rd Latin American (2006) 34–40
3. Harrison Jr, C., King, R.: Folded dipoles and loops. Antennas and Propagation, IEEE Transactions on [legacy, pre-1988] **9**(2) (1961) 171–187