BirdBuggy – Autonomous Mobile Parrot Perch

Andrew C Gray
University of Florida
Machine Intelligence Lab (MIL)
Gainesville, FL 32611
+1 319 541-1182
viron11111@ufl.edu

Dr. A. Antonio Arroyo
University of Florida
Machine Intelligence Lab (MIL)
Gainesville, FL 32611
+1 352 392 2639
arroyo@mil.ufl.edu

Dr. Eric M. Schwartz
University of Florida
Machine Intelligence Lab (MIL)
Gainesville, FL 32611
+1 352 392 2541
ems@mil.ufl.edu

ABSTRACT
Most docking methods involving a small robot and a station require the docking station to communicate with the robot. Few robots are able to dock with a passive base station. Even fewer can also be "driven" by a bird. This paper presents a novel way for parrot-guided transportation and autonomous docking using range lights and a web camera.

Keywords
Computer Vision, Orientation, Threshold, Range Lights, Animal Intelligence

1. INTRODUCTION
Pepper, an African Grey parrot, developed a bad habit. Whenever the bird was alone, he produced terrible shrieks, annoying his owners.

To combat the shrieks an automated sound-triggered squirt gun (Figure 1) was created to deter the bird from shrieking.

Figure 1. Sound activated squirt gun.

At first, the squirt gun proved effective. The bird's screaming would cease after the squirt gun was fired. However, the effectiveness of the water gun did not last. Pepper began utilizing the water as a bird bath and continued to shriek. Next, a remote controlled noise maker was placed near the bird's cage (Figure 2).

Figure 2. Remote control rattling device.

Whenever the bird shrieked, a user could press a button to activate the noise maker, startling the bird. This too did not last as the bird became accustomed to the rattling of the noise maker.

Realizing that the bird's shrieking was due to his anxiety of being left alone, it was concluded that a better option was to allow the bird to roam around the house under his own control. However, because of the messes the bird leaves behind and the possibility of the bird getting stepped on, roaming the house un-attended was not a viable solution. If Pepper could be placed on a mobile platform that would move about the house, the shrieks could potentially stop. Thus the idea for the "BirdBuggy" was born.

The original BirdBuggy was an open-loop, two-wheel drive, tri-cycle platform powered by a 12V battery and controlled by a microcontroller (MC) (Figure 3).

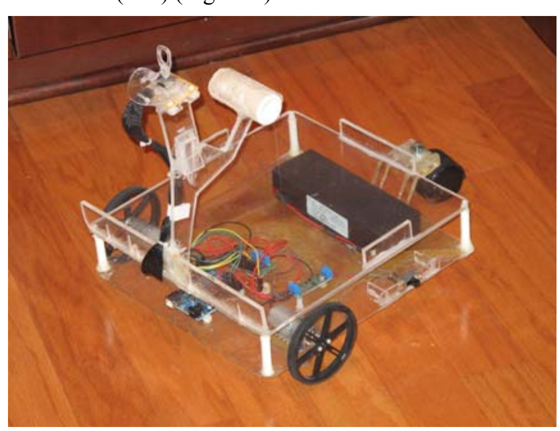

Figure 3. The original BirdBuggy.

The concept was relatively simple; the parrot is placed on the perch and interacts with the Buggy through a joystick. The movement of the joystick directs movement of the cart. With no sensory feedback, the original vehicle had several deficiencies: it was unable to drive in a straight line and it could not avoid obstacles. The goal of the new BirdBuggy was to solve the previous problems and to add some additional functionality to the platform.

This report reviews the construction and details of the new BirdBuggy autonomous parrot perch. The buggy operates in two modes: parrot slave (PS) and return to base (RTB). In PS, the parrot is in control. Moving the joystick will result in movement of the perch. Once the parrot is finished driving, RTB mode is initiated via remote control.

2. BIRDBUGGY SYSTEMS

BirdBuggy uses an ARM processor for the high level decisions and two MCs for low level decision making. Shaft encoders, bump switches, IR range finders, a joystick, and a web camera provide information for the ARM processor to make decisions. The block diagram in Figure 4 shows all of the components and their communication paths.

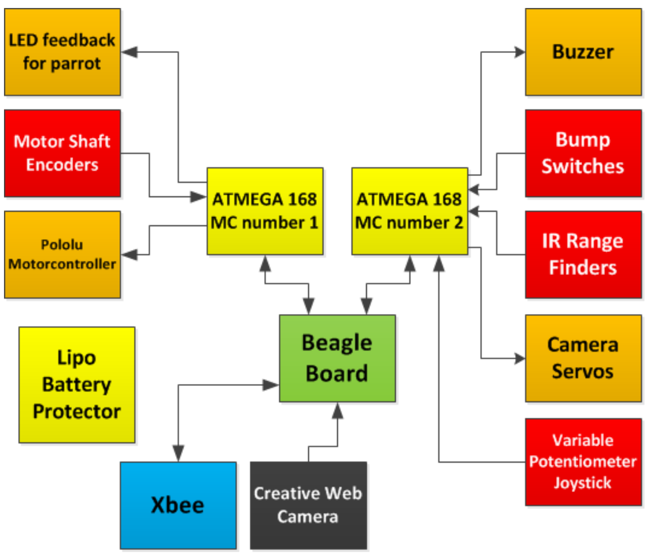

Figure 4. BirdBuggy system diagram.

2.1 Integrated systems

Computer vision requires a great amount of computer resources, i.e., it is processor intensive. Using a very fast computer would result in a shorter image processing time. However, fast computers require large amounts of energy. On a small robot, where power consumption is important, having a fast computer is usually not an option. Therefore, a compromise between processor speed and energy consumption had to be made. The BeagleBoard-xM [2] (Figure 5) with its ARM-A8 processor, can handle image processing while only consuming approximately 10 Watts.

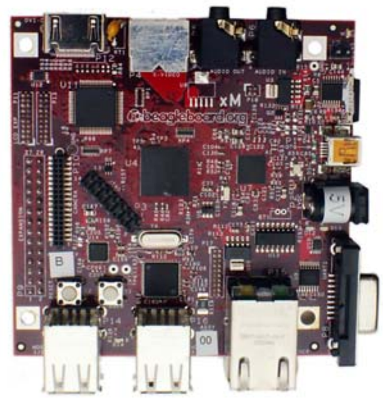

Figure 5. BeagleBoard-XM computing board.

To reduce the amount of processing required by the BeagleBoard, two Atmel ATmega 168 MC were selected. The ATmega 168 chip has 28 pins with two interrupts. They are fast and inexpensive. One MC was assigned to control the motors while the other was the designated sensor monitor and servo controller.

The camera was interfaced with the BeagleBoard using a USB cable. Because of the shape of the camera and the position of the lens, the camera was mounted vertically as shown in Figure 6. The different view from the new orientation was corrected using software.

Figure 6. Web camera mounted sideways.

2.2 Mobile Platform

For simplicity, the robot uses two wheels and two motors for propulsion with two castors for platform support. The web camera is mounted to the robot through two Hitec servos allowing two additional degrees of movement for searching.

Micro-controller one (MC1) was interfaced with a Pololu motor-controller through several pins. To ensure that the robot moved in a straight line, interrupt pins from MC1 were attached to the shaft encoders on the motors. MC1 performed proportional integral derivative (PID) control to ensure the motors turned at the same rate. MC1 also illuminated several LEDs for testing purposes to signify when the base station had been sighted. MC1 was able to communicate with the BeagleBoard through a USB port.

The camera pan and tilt system consisted of two servos and an L-bracket. A small HI-TEC servo provided tilt and a larger HI-TEC servo pan. Able to pan and tilt up to 180 degrees, the platform provided a wide range to allow a large viewing angle for the camera.

To allow the bird to remain on the buggy while it moved, a small perch was attached to the top front of the rover. Silicone caulk was added in a rib-like fashion to allow the parrot's claws to establish a secure grasp.

Inspired by navigational range lights, the docking station consisted of two spherical bulbs painted neon green and neon orange. The green bulb was placed in front of the orange at a lower height. Whereas the orange bulb is in line with the green light but positioned higher (as shown in Figure 11).

2.3 Power and sensors

The BirdBuggy is powered using a two cell lithium polymer (lipo) battery. Due to the dangers of inadvertently damaging lipo batteries by prolonged use, a lipo battery monitoring board was created and installed into the BirdBuggy. The lipo battery protector monitored both cells to ensure that neither was drawn too low. Should the battery voltage go below a selected threshold, the circuit removes power from the entire Buggy to protect the battery.

For obstacle detection and avoidance, several sensors were utilized. Two horizontal arms connected to two switches were attached to the front of the rover (Figure 7). Upon physical contact with an obstacle, the switch was activated by completing an electric loop. The arms detected obstacles that were too small or out of the view of the IR sensors. For longer range obstacle avoidance, two IR sensors [3] were mounted to the front of the chassis. These sensors emit IR light and detect the amount of IR light reflected off of an obstacle. The sensors were used during slave mode to prevent the parrot from hitting an obstacle and during docking mode to detect the base station.

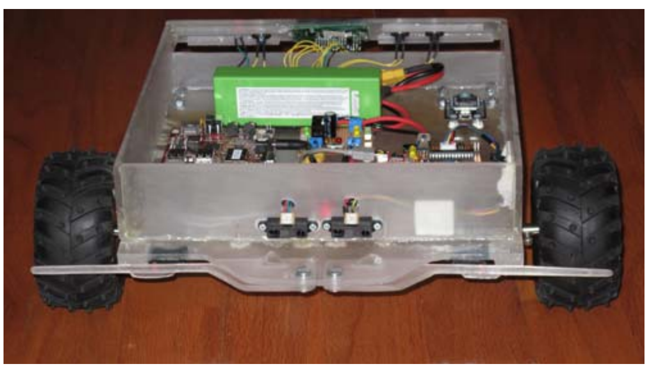

Figure 7. Contact bumpers and IR range sensors mounted to the front of the chassis.

Each IR sensor was connected to 5V and ground. The analog signals were monitored by micro-controller two (MC2) on two ADC pins. MC2 communicated with the BeagleBoard via USB across a FTDI chip. MC2 also controlled the pan and tilt servos for the web camera.

For visual identification and base station position orientation, a Creative Live! cam chat HD was used [4]. The camera provided 640 by 480 pixel image resolution and a frame rate of up to 30 frames per second (FPS). However, the BeagleBoard had difficulty with the resolution, resulting in a lower FPS (~1 FPS). The low FPS severely slowed the Buggy because the docking pace was determined by the speed of the image processing. To improve the slow speed, during search mode, the resolution was reduced to 320 by 240 (~3 FPS). During the approach phase, when movement speed was reduced, the image resolution was reduced further to 160 by 120 in order to accelerate the frame rate to around 5 FPS. The camera was mounted vertically (90 degrees clockwise) allowing the pivot point to be closer to the perch stem.

The joystick was made from two 10kΩ 2-axis variable potentiometer, utilizing 5V, ground, and two wires for analog voltage readings. A loop shaped piece of polycarbonate was attached to the top of the potentiometer for the parrot to manipulate with his beak (Figure 8).

Figure 8. Polycarbonate joystick activated by the bird.

Values from the analog to digital converter (ADC) pins were used to determine the direction of desired motion. Red and green LEDs were mounted next to the joystick in order to provide visual feedback for the bird.

3. Behaviors

3.1 Parrot slave mode

When first powered on, the buggy defaulted to parrot slave mode. During this mode, the parrot was in command of the robot (Figure 9).

Figure 9. Bird operating the BirdBuggy in slave mode.

If the parrot moved the joystick forward or backward, the buggy moved several inches forward or backward. Moving the joystick left or right resulted in a ten degree turn in the corresponding direction. For visual feedback, a red LED was illuminated during left turns and a green was illuminated during right turns. While the parrot was driving, MC2 monitored the obstacle detection/avoidance sensors. If one of the sensors detected an obstacle, the buggy either turned away or prevented forward movement. Once the bird has been removed, the human could activate the RTB mode via wireless communications using XBees.

3.2 Autonomous docking mode (RTB)

Once activated, the rover initialized the camera and began a sweeping search using the pan and tilt servos. During the search mode, the camera panned clockwise in eight degree steps until either it finds the base station or it reached a maximum angle. If the angle is reached, the direction is switched to counter-clockwise and the camera continues to pan in eight degrees steps until the base station is reached.

The process to identify the base station required images from the camera to be converted from red, green, blue (RGB) to hue, saturation, and value (HSV). Identifying particular colors under different lighting environments is more consistent using HSV than using RGB [1].

Next, two thresholds were applied to the captured image: HSV for neon green and HSV for neon orange. The BeagleBoard then measures the area of the green and the area of the orange. In order to be considered the docking station, both the green and orange area must be greater than a specified amount. If only one color or neither color is larger than the required threshold, the camera continues to search for the station. When the area for both matches or exceeds the minimum, further calculations are performed on the image. An image of both thresholds can be seen below (Figure 10).

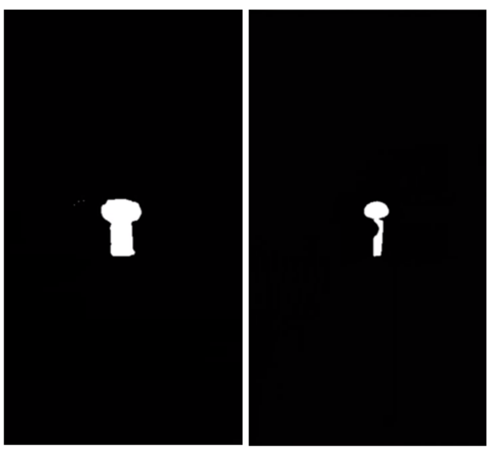

Figure 10. Threshold images of the green, left, and orange, right, spheres.

When the BeagleBoard identified the base station, it calculated the center of the green and orange globe relative to the camera image. The center of the orange globe is subtracted from the center of the green globe and the result is saved as the variable *diff*. If the difference between the two was too great, the image is disregarded and the rover continued to search. If the distance is acceptable, the rover then checks whether *diff* is positive or negative. Having a positive *diff* means that the green sphere is to the right of the orange (Figure 11).

Figure 11. Camera viewing a positive diff between the orange and green spheres.

A negative *diff* indicates that the green sphere was to the left of the orange.

With the centers of the globes determined, the rover then checks if the green sphere is located in the center of the image. If the green globe is not centered, the BeagleBoard uses the pan and tilt system to move the camera until it is centered on the green sphere. This process is conducted prior to every rover movement.

After the difference has been calculated, the rover then looks at the area of the spheres to estimate the distance to the base station. If the area is large, the rover assumes that the station is close and increases the tolerances for determining whether the globes are aligned or not. Having a small area value, the rover assumes that the station is far away. The tolerances are decreased in determining if the globes are aligned.

Now that the rover has centered the camera on the green sphere, determined the orientation, difference of the station, and estimated the distance, the rover begins moving to align itself for an approach.

If the distance to the base station is estimated as far, the rover will pivot about the camera in a way that would allow a perpendicular approach. First, the relative bearing of the station is determined. If the bearing is not within 25-55 degrees for a positive *diff*, or 135-165 for a negative *diff*, the rover turns until the station relative bearing falls within one of the ranges. Once the bearing lies within the correct range, the device moves forward a small distance. *Diff* is recalculated to see if the spheres are aligned with tight tolerances. If the spheres are not considered aligned, the aligning and moving process is conducted again until the globes are considered aligned.

When the BeagleBoard determines that the station is close, the above bearing ranges are changed to allow a shallower approach. The left range field is reduced to 60-80 degrees for a positive *diff* and 100-120 for a negative *diff*. Using the shallower bearing ranges made the approach more direct because of the decreased distance. The rover maintains the relative bearing of the station and proceeds forward until it determines that the two globes are visually aligned.

Once the rover determines that the spheres are aligned, no matter the distance, the rover turns to face the docking station. When the

BeagleBoard determines that the robot is facing the station, it proceeds forward maintaining the green globe directly in the center of the camera. During the approach, the rover monitors the two IR sensors in the front to determine the range to the docking station. When the docking station is approximately six inches away, the robot stops and announces its completion with an audible tone.

4. RESULTS

4.1 New features

Using the shaft encoders, the buggy was able to maintain a very straight line while driving forward and backward. Increasing the accuracy of the buggy allowed the parrot to drive better and learn quicker.

The new stand was much better than the previous stand. On the original BirdBuggy, sandpaper was used to improve the grip of the parrot. However, over time, the sandpaper was worn away and became smooth. This resulted in several accidents in which the parrot was thrown off the buggy. Applying silicone to the new stand resulted in a much more stable bird.

To prevent the parrot from escaping the buggy, the sensors were mounted to the front of the rover. Obstacles were detected by the sensors preventing some escapes. However, due to the physical location of the sensors, not all obstacles were discovered. Objects that were above the sensor field of few were not detected. While better than the original, the parrot driving the new BirdBuggy still required supervision.

4.2 Docking mode

Upon activation of autonomous mode, the camera consistently discovered the base station with a maximum range of ten feet (depending on the lighting), figure 12.

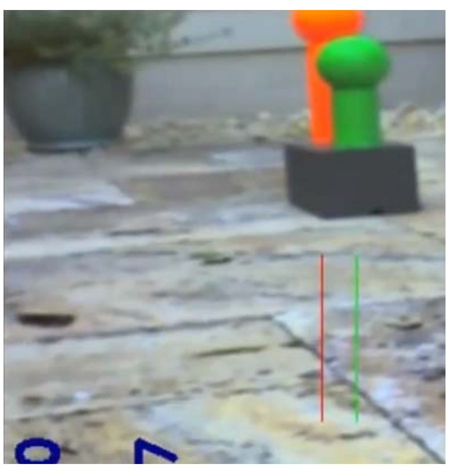

Figure 12. Base station detected with bearing displayed.

Because of the BeagleBoard's slow processor, the image was halved from 640x480 to 320x240 during search mode. The smaller resolution allowed for faster image processing and did not adversely affect the detection range. During the approach phase, the image was further reduced from 320x240 to 160x120. During the approach phase, the movements of the rover were slower and more deliberate to ensure an accurate dock. Having an even smaller frame reduced the time required for the robot to dock while maintain an accurate approach.

If the robot was able to detect the docking station, there would be a very high probability that it would dock correctly. Distances greater than ten feet made detecting the base station difficult for the rover. At that distance the areas of the orange and green globes were close to the minimum detection size. Often the globes were disregarded as noise. Lowering the detection minimum introduced more false detections.

Having poor lighting resulted in smaller threshold areas preventing detection. The sun provided the best lighting as long as the direction of light was from the sides or front of the docking station. If the sun was directly behind the docking station, the web camera automatically corrected the image by decreasing the brightness resulting in smaller detected sphere areas.

5. CONCLUSION

BirdBuggy provided a way for the bird to move about the house with minimum human interaction. The buggy was also able to autonomously dock with a base station.

5.1 Improvements

The greatest limiting factor of the robot is the detection range of the docking station. An improvement to this handicap would be to add omni-directional IR transmitters to the docking station and IR receivers on the rover. While the IR sensors do not help with the autonomous docking orientation, they provide a longer detection range than the camera. When the robot's autonomous mode is activated, the rover would begin searching for the station using the camera and the IR sensors. Assuming the station is greater than ten feet away, the rover would first pick up the IR transmitters. The individual sensor detecting the IR light would give a rough bearing for the rover. Next the rover would turn towards the light, and drive towards the light source while using its camera to search for the spheres. Once the spheres are detected with the camera, the robot would commence the docking mode describe above. Using IR transmitters could extend the robot's detection range to greater than thirty feet.

Currently, if the rover does not detect the two globes, the robot will remain indefinitely in its original location. Having the rover conduct a search pattern would increase the probability of detecting the docking station. Using an expanding spiral pattern, the robot would initially conduct a sweeping search with the pan and tilt system. If the station was not detected, the rover would move forward a short distance, turn ninety degrees and conduct another sweeping search. If the station was still not detected, the rover would move forward a short distance, turn ninety degrees, and conduct a third sweep. This process would continue, lengthening the forward driving distance after two sweeps, until the base station is detected. Looking down on the robot, the robot would appear to be making a square-like expanding box similar to a spiral.

The current design of the rover does not utilize the bump sensors or IR range finders for obstacle detection during the docking mode. Should an obstacle be located between the rover and the station, the rover would either push it out of the way or become stuck trying to move towards the docking station. An improvement allowing the rover the ability to move around obstacles using the bump sensors would prevent miss-aligned docking.

6. ACKNOWLEDGMENTS

Thanks to Dr. A. Antonio Arroyo, Dr. Eric M. Schwartz, Tim Martin, and Josh Weaver.

7. REFERENCES

[1] Gary Bradski (2012, May 08). OPENCV Wiki. *Willow Garage* . Retrieved October 24, 2012 from http://opencv.willowgarage.com/wiki/

[2] "BeagleBoard" Available: http://beagleboard.org/ [Accessed: 19-Sep-2012].

[3] "Sharp GP2Y0A21YK" Available: https://www.sparkfun.com/products/242 [Accessed: 7-Sep-2012].

[4] "Creative Live! Cam Chat HD Webcam" Available: http://us.store.creative.com/Creative-Live-Cam-Chat-HD-Webcam/M/B004431UBM.htm [Accessed: 03-Oct-2012].