

### Grow-out House Robotics

Poultry grow-out-houses require a significant amount of labor for carrying out tasks such as monitoring flock health, removing mortality, and picking up floor eggs. In addition to the labor issues, there is also concern about contamination and diseases being introduced by people. Therefore, the industry is very interested in technologies that would allow to reduce or eliminate the need for people to enter these facilities. This sounds like a ripe robotics application to us!

Over the past two years, the team at Georgia Tech Research Institute has designed, developed, and tested a fully autonomous robot capable of navigating through a dense flock of birds in a commercial house. Algorithms were designed to allow the robot to physically interact with the birds, such as nudging them to encourage them to move out of the path of the robot. After playing chicken, if the robot is still unable to continue, it will dynamically plan an alternate route around the offending bird. In addition to interacting with the birds, an approach for identifying, and automatically removing floor eggs was developed using deep learning and the tensor-flow library. Image processing routines for identifying chicken in both 2D and 2D data were also developed. All of this functionality is run on an Nvidia Jetson TX2 platform.

Our robot is assembled using affordable and commercially available components. These include an NVIDIA Jetson TX2 board, Microsoft Kinect v2 RGB-D camera, uArm Swift — a 4DOF robotic arm with a payload of 500g, and a Raspberry Pi board used to interface with the arm. **Error! Reference source not found.** gives an overview of the current system architecture as well as the external view of the rover with the robotic arm mounted.

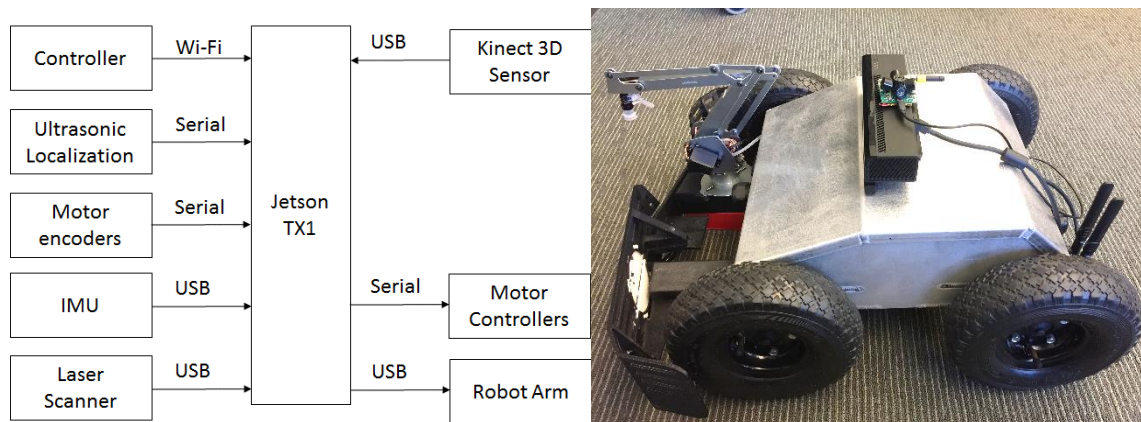


Figure 1- GOHBOT System Overview

The robot's regular behavior consists of navigating the house between a set of waypoints and involves nudging the chickens to clear the path. Egg detection is designed to run continuously during the navigation and trigger egg-picking behavior if an egg is identified. As a vision sensor we utilize Kinect v2, mounted on top of the robot, with open-source drivers.

Our object detection package is based on an implementation of Faster-RCNN approach using TensorFlow library. This approach integrates object proposals, classification, and bounding box regression into a single network, resulting in tight detection boxes around the objects (see **Error! Reference source not found.**). One notable issue while working on mobile platforms is limited computational resources and memory. To address it, we use a small VGG-M network architecture that

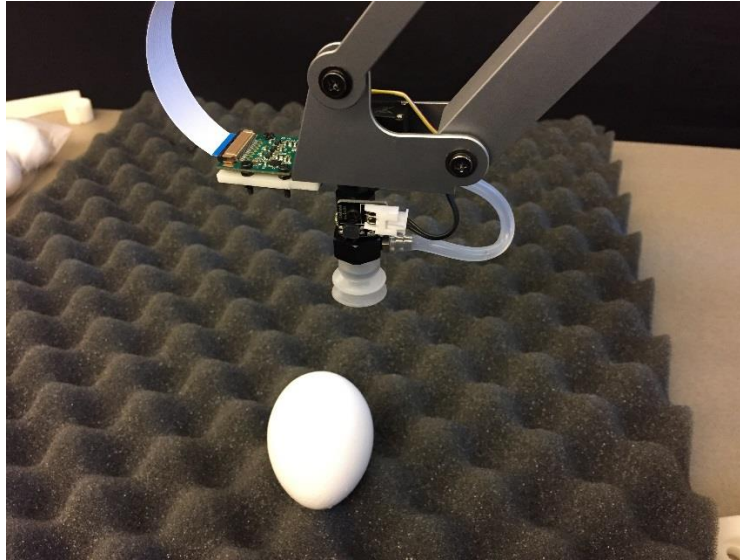
fits into Jetson memory during the runtime. The network is pre-trained on the large ImageNet dataset of generic objects before being fine-tuned to our small domain-specific dataset with about 100 instances.



*Figure 2 - Egg and chicken detections using Deep Learning framework*

We perform detection only using RGB channels and lookup the object's center location in the registered point cloud to get the position of the object with respect to the robot. Once the egg is detected then issue a new navigation goal such that the robot stops with the egg being within robotic arm's workspace. To achieve that we apply a fixed offset to the egg position to get a new goal with respect to the robot end effector. To compute the goal orientation we use the direction vector between the goal position and the original egg location. The new 6D pose is then sent to the planner and the robot drives up to the egg with the final orientation facing the object.

The next stage of the process starts when the robot has approached the egg and has it within its robotic arm reach. To precisely localize the egg we use a Raspberry Pi camera attached to the tip of the arm. Our algorithm utilizes OpenCV blob detector to identify a white blob of elliptical shape and extract its center. We use this output to continuously adjust the position of the arm to gradually center on the egg. Current controls of the arm are implemented using uArm's Python API. Once the distance (estimated in the image space) from the tip to the egg is within predefined threshold we begin to move the tip vertically towards the egg. We continue to descend until the tip's pressure sensor is activated meaning it is pushing against the egg. At that point the suction cup is activated and the robot lifts the egg and places it in a basket mounted to the robot using a predefined motion.



*Figure 3- The robotic arm with a suction cup and attached camera*

Top date, the robot has operated for over 40 hours of fully autonomous behavior in test chicken houses. The image below shows one of these tests and illustrates the waypoint pattern given to the robot to follow. The system was also demonstrated finding, navigating to, and successfully picking eggs from the floor in a lab environment.



*Figure 4- Robot and path for testing with chickens*