Semi-autonomous Warehouse

Khalid Alzuhair, Shiyu(Shirman) Xia, Yikun Li, Abby Xu

University of Washington, Department of Electrical Engineering

Introduction

Fizikl, Inc is a company that uses an Unmanned Aerial Vehicle (UAV), shown in **Figure 1**, to count inventory in warehouses. The goal of the company is to replace the manual inventory counting method that most of the companies use. Our team worked on the barcode detection and scanning process, as well as the micro-guidance for controlling the UAV. The first step to achieve an autonomous warehouse is to pinpoint the UAV's waypoints and designate its flight plan. Then, the UAV will take pictures upon arrival at each destination point after centering itself using the fiducial markers. The distance between the camera and the shelf is carefully set to a fixed value to capture all the items on the shelf. The pictures will then be sent to a ground station where the barcode extraction and counting will happen. The barcode scanner algorithm will also output the location on the pictures where it finds barcodes. For the upcoming sections of this poster, we will explain how, over the past six months, we have successfully built the barcode scanning and micro-guidance systems for the project.

Figure 1. The Intel UAV

Methods

1. **Barcode Scanning and Extraction:**

The first version of the counting system depends on UAV taking pictures upon arrival at a destination. The UAV first will stop at the destination and the micro-guidance system will start working to center the UAV (see next section for micro-guidance). Once the UAV has centered itself, the micro-guidance system will send out signals for the camera to take pictures at the current location. The UAV that we use is the Intel Ready To Fly drone, shown in **Figure 1**. It has two cameras on board - one RGB 2 megapixel camera and one 8 megapixel camera. The 2 megapixel camera is used for autonomous flight, leaving the 8 megapixel camera for taking pictures. However, due to difficulties from the Intel side, we have not yet been able to use the 8 megapixel camera at its full potential. While waiting for Intel to solve the camera driver issues, we have decided to mount a GoPro camera on the drone to take over the 8 megapixel camera. The barcode system can be divided into two sections: barcode scanning and barcode extraction.

i) Barcode Scanning

The GoPro camera takes pictures and sends it back to the UAV ground station after being signaled by the micro-guidance system. The GoPro camera was originally designed to be a wifi host where when turned on, it spreads its own signals and a computer can connect to the GoPro camera's wifi. Because we need the ground station to be connected to the internet and the UAV has to be connected to the same network for various signals to communicate with each other, we have turned the GoPro from a Wifi host to the Wifi client where it now connects to the same network that the ground station and the UAV are connected to. The GoPro then sends back the pictures that it takes back to the ground station and the script we wrote for processing the pictures will take the pictures and analyze them. The scanning is done through a barcode reading software development kit where it takes in pictures and outputs the type of the barcode, the barcode number, and the location where the barcode is located in the picture.

ii) Barcode Extraction

After first step, the barcode scanning information is outputted to a destination file. The contents of the file are then processed for our internal database. The database we have is built on SQL. It keeps track of the number of barcodes and the barcode numbers for reference. We also store the images that are processed previously so that when discrepancies in numbers occur, we can refer back to the pictures; this is also for quality insurance (we can randomly sample picture to determine the accuracy of our barcode scanning and extraction system. There will be a window display that keeps updating the number of inventory it counts during a flight.

Methods (con't)

2. **Micro-Guidance:**

The UAV will have two navigation modes to follow: a general navigation mode, and the micro-guidance mode. In the general navigation mode, the UAV will receive waypoints and use the autopilot to travel to each waypoint. Upon arrival to each waypoint, the UAV will switch to the micro-guidance mode. The general navigation mode that uses the autopilot will not be stable enough for scanning barcode. The micro-guidance mode uses computer vision and visual feedback to help the drone stabilize around fiducial markers, in order to avoid shaky and blurry images.

i) Visual feedback

Fiducial markers: Fiducial markers, shown in **Figure 2**, are markers used to report relative position and velocity of the drone. Fiducial markers also have unique identity, therefore, they can also be used to map all shelf locations and record scanning progress. Fiducial markers vary by dimension. Our team uses 5x5 markers, which will give us 1024 different markers to identify shelf locations. Fiducial markers are easy to detect compared to boxes or barcodes.

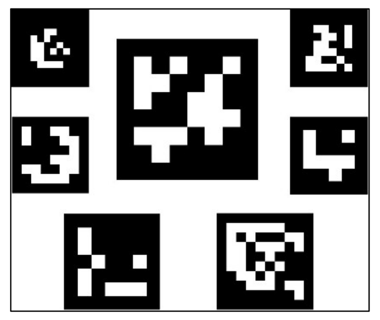

Figure 2. Fiducial Markers

Detection vs. Tracking: Developing perfect detection algorithm is hard. Detection requires specific conditions, including orientation, lighting etc. Detection often loses track of the target. Tracking however, only compares the current frame to the previous frame and finds the most similar object. In order to provide stable visual feedback, detection and tracking are combined.

ii) Control

Using the relative position visual feedback from the fiducial markers, our team has developed a PID controller that commands linear velocities to the UAV to center the UAV around the fiducial marker, as can be seen in **Figure 3**. Our team first developed a P controller, which had overshoot problems as expected. Next, we developed a PID controller, which we tuned. After the PID tuning, our system follows the required stability specifications for the barcode scanning. All of the simulation tests have been done in Gazebo, which is a robotics simulation environment. **Figure 4** and **Figure 5** show the simulation setup and the view of the on-board camera from a simulation test.

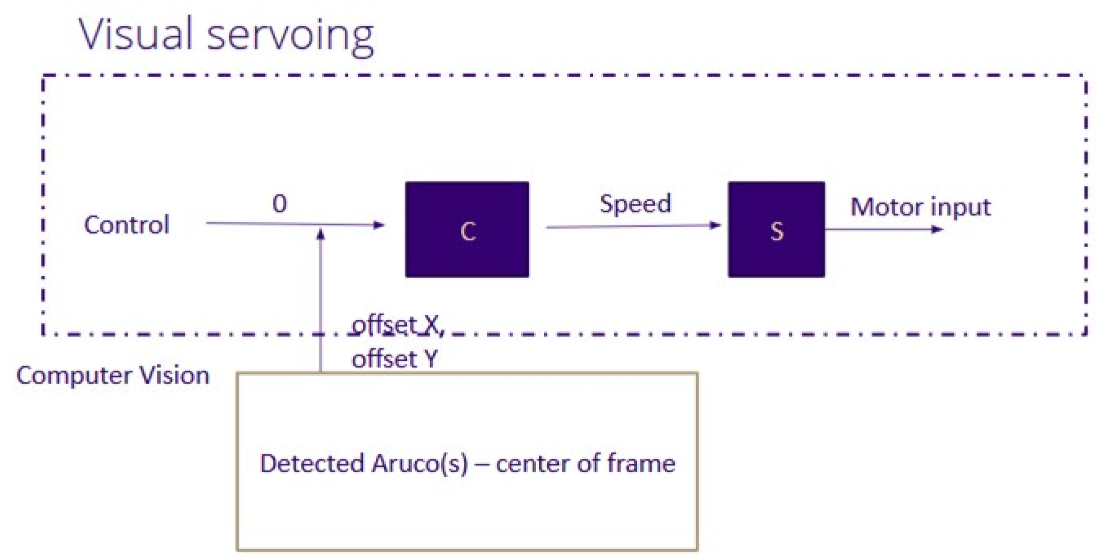

Figure 3. The control diagram of the system

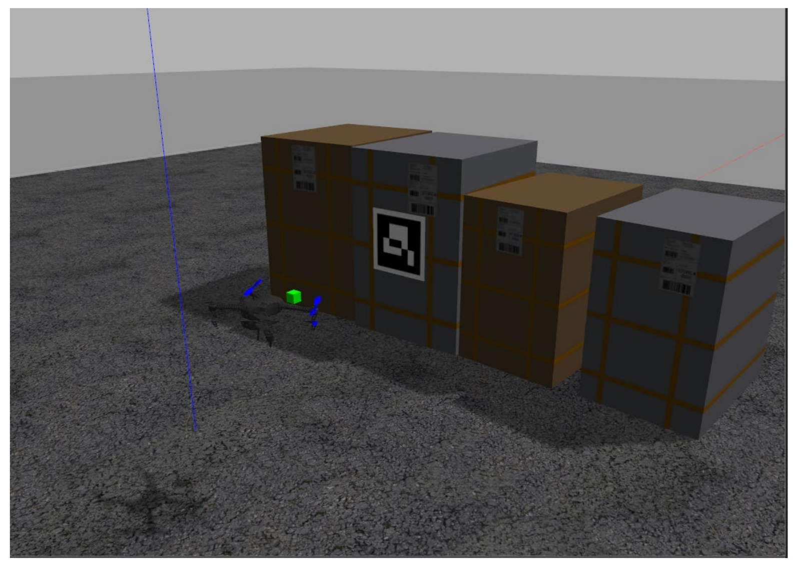

Figure 4. Gazebo Simulation view of UAV

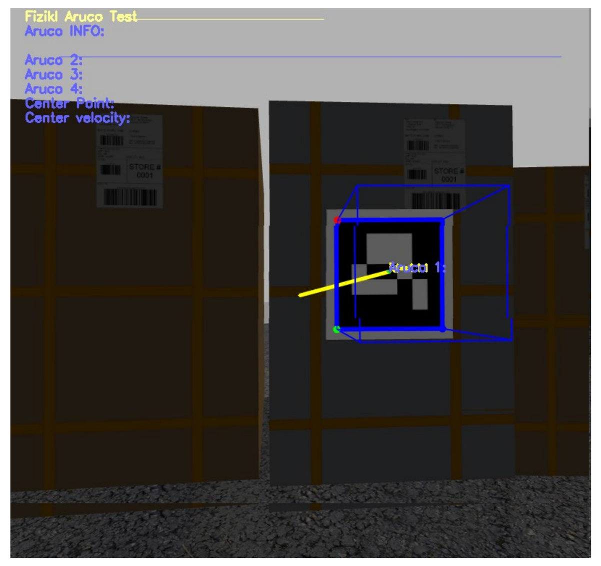

Figure 5. On-board camera view

Results

1. **Barcode Scanning:**

The barcode scanning scanning and extraction system was built using the system shown in **Figure 6**. The GoPro can be triggered when the UAV is centered at the shelf location. Our barcode reader software is able to recognize decent size barcodes in the photo taken by the Gopro. Our program can correctly update the recognized barcode information including barcode numbers, images, and product information in our local database and update the counts of products at the output console.

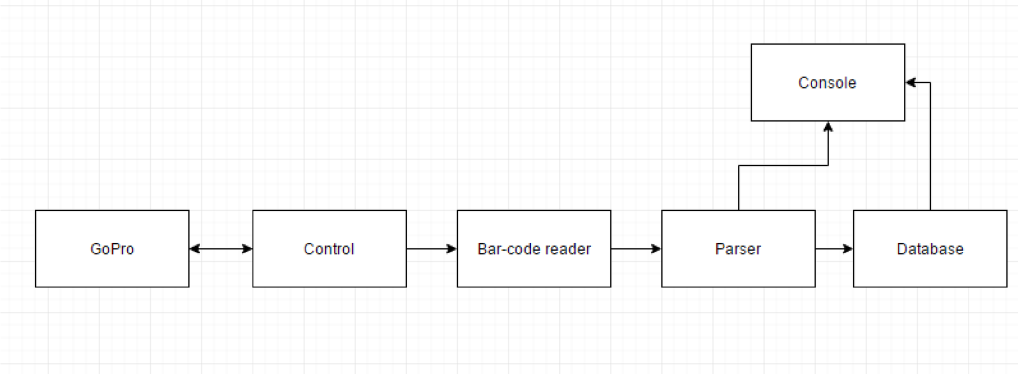

Figure 6. Barcode Scanning and Extraction System

2. **Micro-Guidance:**

To make sure that our algorithms are functioning properly, we are doing our Software In The loop Tests (SITL) using Gazebo and the Robotic Operation System (ROS). The setup for the simulation system is shown in **Figure 7.** The UAV is part of the Gazebo box which publishes the camera image to the fiducial marker detection. After the detection of the marker, the marker's offset is then published to our controller node. The controller node uses the offset to commands velocities to the UAV using mavros. Along with the UAV and the fiducial marker, we have added some shelves in the simulation environment to simulate a real warehouse. By interfacing ROS with Gazebo, we were able to test our controller. The tuned PID controller showed very successful results by moving the UAV such that the fiducial marker is at the center of the camera view stably.

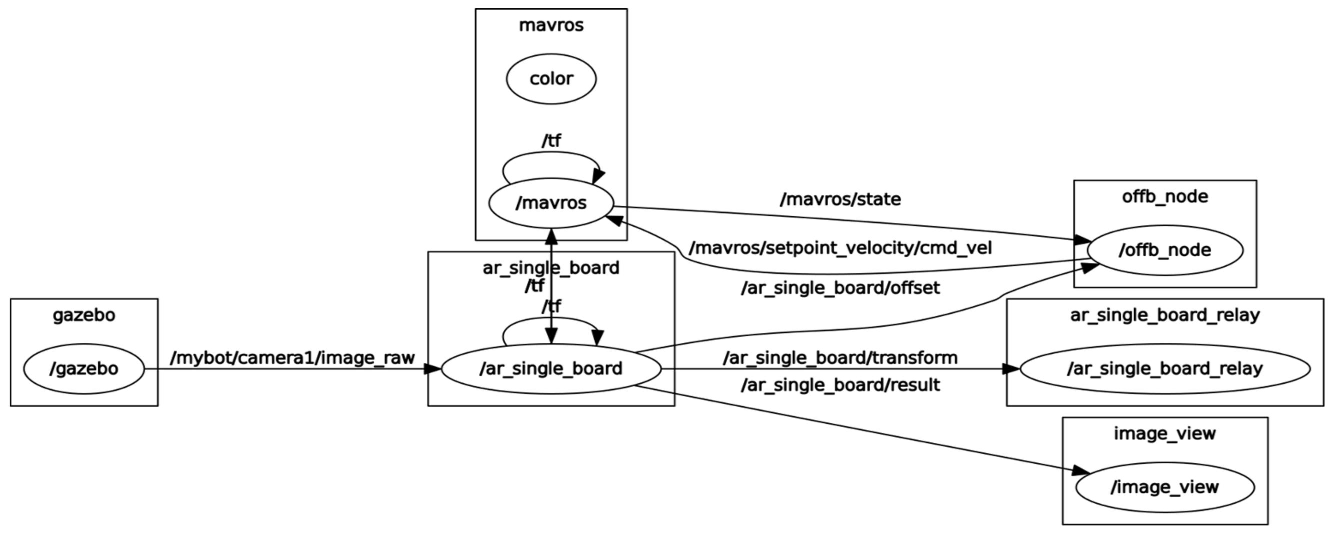

Figure 7. The simulation environment system setup

Conclusion

Our team has successfully built the barcode scanning and the micro-guidance systems for this project. Our team was able to use a GoPro camera to replace the 8 megapixel camera that did not provide full resolution, due to issues from the Intel side. We can now use the GoPro to take pictures, and then send these pictures to a barcode scanner that outputs the information for the barcode. Then these information are stored in a database. We have also developed a PID controller that uses visual feedback with tracking to give the UAV relative position information. We have tested the SITL of our micro-guidance controller in Gazebo. Moving forward, we are planning on conducting a flight mission, where we simulate a real flight by combing the general navigation mode with the micro-guidance mode. For the barcode system side, the team will be working on developing a GUI for the barcode scanning and extraction, as well as, integrating the shutter trigger with the micro-guidance system.