CS/ECE528 Project Milestone Report

Elaine Smith, elaines@colostate.edu, 832556432
Nathan Kopacz, ndkopacz@colostate.edu, 832987226

2024

Paired Automated Image Range (PAIR) - Project Milestone Report

1 Introduction

Our project addresses the challenge of drone movement guidance in GPS-denied environments. The current reliance on GPS for drone navigation is limited by its 1-meter accuracy and frequent unavailability in indoor or urban settings. This project aims to overcome these limitations by using a machine learning model that processes image pairs to determine relative positioning. Our approach will enable drones to navigate indoors and in areas without GPS by using relative movement commands.

2 Problem Statement

The primary goal is to develop a system for indoor drone navigation based on visual data, replacing the need for GPS. We will use image pairs captured at specific intervals to determine the relative positioning of the drone. The dataset will consist of both synthetic images generated through simulated environments in Unreal Engine 5, as well as real-world data collected manually using a Raspberry Pi with a Pi camera. The expected result is a machine learning model capable of accurately estimating the six degrees of freedom (x, y, z positions, roll, pitch, and yaw) between two consecutive images, with evaluation using RMSE for positioning accuracy.

3 Technical Approach

We are considering two main approaches to solve this problem:

1. **Custom Neural Network Architecture**: A custom network will take two images as inputs and output the six degrees of freedom values, with convolutional layers for feature extraction and fully connected layers for regression.
2. **Modified YOLO-based Model**: Another approach involves using two YOLO networks connected by fully connected layers, allowing the model to leverage YOLO's fundamental understanding of objects already at scale.

For our synthetic data, we are using Unreal Engine 5. In this game engine, we are able to generate a scene of a room using real life objects. We can then move a camera around the room, capturing images and calculating the relative distance between camera positions. Once automated, this can generate a enormous dataset of a single room. Due to timing constraints on our project, it is likely that we will only build and generate one scene. However, for a more robust model that can handle different environments it would be an easy step to alter the lighting of the room to simulate different times of day. Additionally, we could build additional scenes of differing environments including city scapes, urban environments, office buildings, rooms of a house, etc.

Our real life data is being collected with an abundance of caution so that we can get accuracy metrics on the accuracy of our model. To this end, we are modeling and 3-D printing a case for the pi and its camera as well as mounting brackets for the case. The mounting brackets for the case are also 3-D printed at precise angles. This will allow us to take an image, change the bracket for a new angle, then take another image. This will give us precise data points for various rotational angles. For movement in the X, Y, and Z direction, we will we using physical measurement tools which should give us accuracy up to 1 cm. This should be more than enough control for the use case of macro movements on a drone.

Another benefit of the Raspberry pi system is that it will enable a live demonstration through a simple web portal and graphing of the model's expected position. It is critical that the pi is able to run the model in real time, so part of our analysis will be in reporting on the latency of the model on our edge device.

4 Intermediate/Preliminary Results

So far, we have successfully built a realistic scene in Unreal Engine 5 and begun the scripting for the synthetic data. Image's can be captured in the correct resolution and the camera can move in a random distance from its original point. The distance, pitch, roll, and yaw are saved. There are a few small piece left in the script before we can do a major run to generate our dataset.

We have bought and received our Raspberry Pi 5 and have Ubuntu 24.04.1 flashed and set up with a temporary web portal in place for collecting data. A case has been printed for the pi and the different configurations for the brackets have been modeled and are slowly being produced.

Conclusion and Next Steps

Our project is on track to achieve indoor navigation using relative positioning. In the next phase, we will focus on:

- A full run of our data generation script.
- Creating our model and training on the synthetic data.
- Begin collecting real world data using the 3-D printed bracket system.
- Creating a testing pipeline using synthetic data and real data as well as analyzing the latency of the different model types on the Pi.
- Creating the live demo script to run the model and plot expected positions.