HACETTEPE UNIVERSITY
FACULTY OF ENGINEERING
Department of Computer Engineering

Active Object Tracking on a Drone

Project Owners: Esad Boran & Yusuf Koca & Mustafa Emir Peker
Project Advisor: Dr. Özgür Erkent

Introduction

The rapid growth of drone technology has created new opportunities for research and practical applications, particularly in the realm of autonomous systems. One of the critical capabilities that enhance the functionality of drones in these applications is their ability to autonomously detect and track objects. This project focused on developing an active object tracking and detection system for drones using the Airsim simulator.

Problem Definition

Our first goal in this project is to create an environment asset on the Airsim Simulator. Initially, we placed multiple vehicles and assets in the environment. After setting up the environment , we implemented multiple vehicle detection and multi object tracking using artificial intelligence models. We implemented a physical tracking algorithm for the drone to follow one selected object acquired from multi object tracking. We also tried to implement a system to seamlessly switch between objects that are followed within the view of the camera.

Methodology

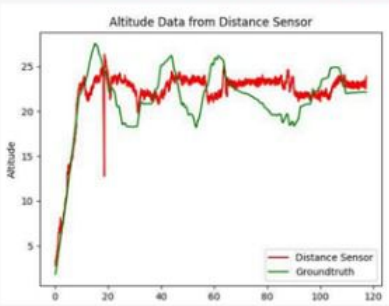
Airsim: AirSim, developed by Microsoft, is a simulator for drones and other autonomous vehicles is used for drone simulation.
Unreal Engine 4: Unreal Engine served as the platform for creating high-fidelity simulation environments.
Datasets: A combination of datasets such as VisDrone and our own custom dataset from UE4 environment is used for training.
Object Detection: Fintuned YOLOv8-s is used for main detection model for tracking by detection.
Object Tracking: For main Trackinh by detection algorithm BoTSrort is used. BoTSORT provides a robust rracking and re-identification with use of, kalman filter, IoU and cosine-distance fusion.

Re-Identification: On top of BoTSORT's re-identification we introduced our own custom re-identification with kalman filters and IoU approach for better results in occlusion.

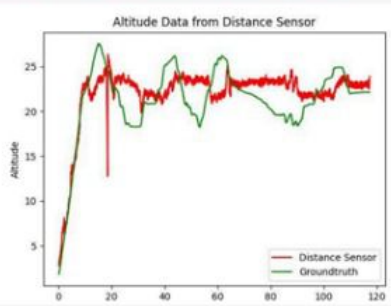
Results

Test Environment: We used a preset track as our test environment with multiple vehicles. Our track includes several curves, uphills and downhills to increase road complexity. It also includes several bridges, trees, road signs to provide full or partial occlusions to push our projects to its limits.

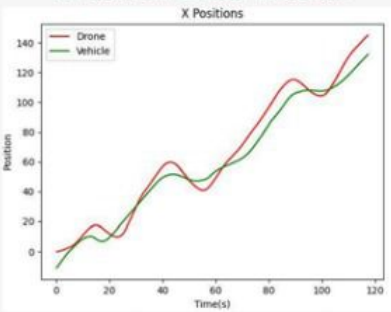
Real Distance vs Camera Distance:



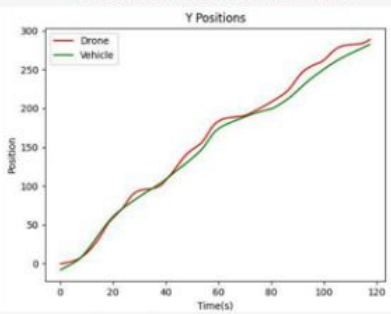
Distance Sensor Results:



X Position Sensor Results:



Y Position Sensor Results:



Demo: https://youtu.be/XB9QJQ73OIs?si=8dktolOVHdDmS6V0

Tools and Datasets

Tools used in this project: Python, Ultralytics, Airsim, Unreal Engine 4, Roboflow.
Datasets used in this project: Visdrone, UAVDT, EXID, Custom Dataset.

References

[1] Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2015, June 8). You only look once: Unified, Real-Time Object Detection.
[2] Aharon, N., Orfaig, R., & Bobrovsky, B. Z. (2022, June 29). BoT-SORT: Robust Associations
[3] microsoft. (n.d.). GitHub - microsoft/AirSim: Open source simulator for autonomous vehicles built on Unreal Engine / Unity, from Microsoft AI & Research. Multi-Pedestrian Tracking.

Information

esatsuphi@gmail.com & kocayusuf095@gmail.com & mustafaemirpeker.mep@gmail.com
https://github.com/BBM479-480-DesignProject/Active-Object-Tracking-on-a-Drone