Enabling Deep Learning based Perception & Control for Aerial Robots

Krishna Muvva

School of Computing & Biological Systems Engineering

Introduction

Advances in deep learning, particularly CNNs, enable robust perception systems for robots. However, CNNs are static, computationally intensive, and present challenges for aerial robots with limited resources.

To address this issue, an adaptive perception approach is adopted. We implemented Adaptive Convolutional Neural Network (ACNN) which can turn on and off its filters in accordance with latency-accuracy requirements.

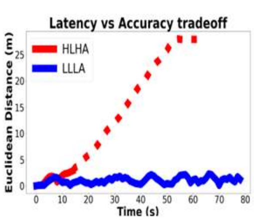
Our approach, tested in a UAS-UAS tracking scenario, shows our system outperforms HLHA and LLLA systems. Major contributions include Adaptive Darknet and a novel Adaptive Perception Controller using TD3.

Problem Statement

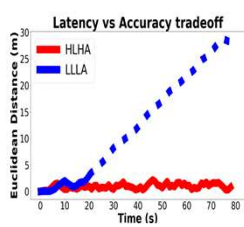
High Latency High Accuracy (HLHA) networks experience significant latency issues, which present a considerable challenge in tracking evasive target.

Low Latency Low Accuracy (LLLA) networks suffer from poor accuracy, which poses significant challenges in tracking targets within complex environments.

Convolutional Neural Networks (CNNs) do not need to operate at full capacity at all times to achieve improved target tracking.



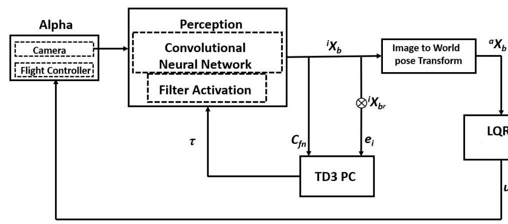
Evasive Target



Dynamic Environment

Methodology

- **Situation aware MPC Controller (SMPC) :** Design of a Model Predictive controller that can produce accuracy and latency requirements in accordance with the situation
- **Adaptive Convolutional Neural Network (ACNN) :** Implementation of a convolutional neural network which can dynamically select the filters based on their significance.
- **Adaptive Darknet :** We have modified the Darknet C code and implemented Adaptive Darknet. Through this one can create their own Adaptive YOLO CNNs.
- **Adaptive Perception Controller :** Implementation of a novel adaptive perception controller with state-of-the-art of reinforcement learning technique i.e. Twin Delayed DDPG (TD3).

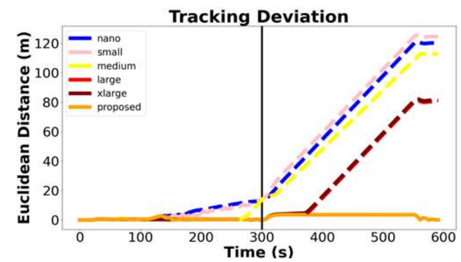


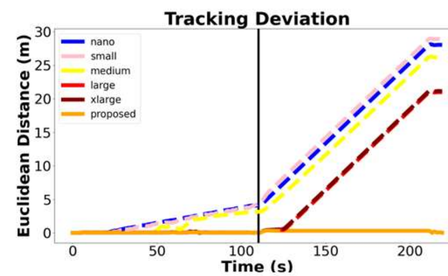
Experimental Setup

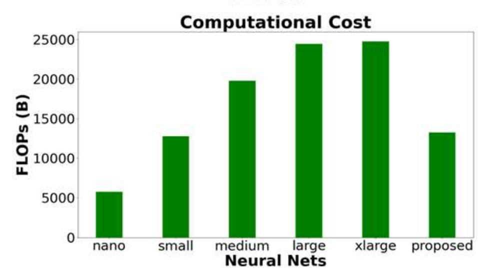
Our experimental setup integrates PyTorch, the ArduPilot suite, MAVROS (ROS Melodic), and AirSim. CNN operations are carried out by PyTorch, while control aspects are handled by MAVROS (ROS package for Micro Air Vehicle Link). The flight controller of the vehicles is handled by ArduPilot firmware. The simulator plant is AirSim, which is a state-of-the-art simulator for ground and aerial vehicles. The physical plant comprises of two UAS equipped with a CubePilot flight controller. Each UAS contains a companion computer board featuring an Intel i5 processor and 16 GB of RAM

Results

- The proposed approach tracks the trajectory successfully, while others fail. Low-latency networks like YOLOv8 nano, small, and medium struggle in complex environments. YOLOv8 large and xlarge fail during evasive maneuvers due to slow inference. The adaptability and control-feedback loop of our approach enable effective target tracking.
- The FLOPs vary slightly in baseline approaches and depend on perception inference and trajectory deviation in our proposed method. Our approach is the only one that successfully tracks the target UAS, with total FLOPs comparable to the YOLOv8 small network.







Conclusions

Our proposed approach adjusts latency and accuracy based on the situation, outperforming baseline methods and being computationally more efficient.