

Zongyao Li

Email: lizongyao000@gwu.edu
Tel: (202)-717-1919
www.linkedin.com/in/ZongyaoLi
Personal blog: lzongyao.com

Education

The George Washington University, School of Engineering & Applied Science Master of Science in Computer Science <ul style="list-style-type: none">GPA: 3.9/4.0	Washington, DC January 2023
Liaoning Normal University Bachelor of Computer science and technology <ul style="list-style-type: none">GPA: 3.2/4.0	Dalian, China June 2019

Research Interests and Main Course

- **Research: Computer Vision, Machine Learning, Drone**
- **Course:**
 - **Machine Learning** 4.0/4.0
 - **Augmented reality (AR) and Virtual Reality (VR)** 4.0/4.0

Publications

- Zhenhao Zhao; Jonathan Lee; [Zongyao Li](#); Peng Wei, **Camera Vision based Perception for UAS Autonomous Landing**, AIAA SciTech 2023

Relevant work experience

George Washington University Research Assistant research content: Computer Vision, drone auto-landing <ul style="list-style-type: none">• Use computer vision models to analyze data taken by drones• Training models on Google Colab• Collect data to build dataset to detect the model's recognition of people and cars during landing	Washington, DC December 2021 - Present
George Washington University Graduate Teaching Assistant <ul style="list-style-type: none">• Collect latest VR/AR technology principles and new applications• Maintain two classes of VR/AR devices, More than 30 devices Model in Oculus Quest 2• Grading of homework exams and answering students' questions for nearly 60 students	Washington, DC July 2022 - Present
Jinyu Media Front-end engineer <ul style="list-style-type: none">• Created web front-end sites on cloud-based servers using only two weeks• Chose cloud server for website, doing daily operation of server, and update website content and perform SEO on website	Dalian, China July 2019 - April 2020

Research and Projects

Research Project: Machine Learning, Computer Vision for Small UAS Advisor: Professor Peng Wei <ul style="list-style-type: none">• Use computer vision to assist drone landing, use computer vision to help in drone landings, helping drones detect people and vehicles near landing pad• Train two models RetinaNet and yolov5 with Visdrone, do Model evaluation, compare and analyze the effect of different pre-trained models• Determine type of data needed, collect hundreds of videos taken by drones for model training, and analysis drones flight data	December 2021 - Present
--	--------------------------------

- Test the effect of drone images at different heights on model detection accuracy
- Deploy the model on a small mobile platform and test the results (NVIDIA Jetson Xavier)

Project One: Machine learning, computer vision to identify dog breeds

January 2022-May 2022

- Research the impact of poorly performing datasets on model training
- Use Stanford dog dataset to distinguish 120 breeds of dog with Resnet
- Make modifications to the existing ResNet model and observe the effect of the modified model

Project Two: Front-end development, Household Goods Management System

April 2021 - June 2021

- Created a website can across multiple platforms with uni-app framework. The website can be use on more than 5 platforms
- Develop a login user system with 4 user group permissions. Different user class different function
- Collaborate with back-end development for testing and data exchange between front and back-end, for nearly 100 API

Project Three: Infrared Active Contour Tracking of Moving Vessels

June 2018 - June 2019

- Manage a mobile platform to support for the operation of infrared modules
- Classify image data and tag data according to the needs of model training
- Manage the connection of infrared devices to portable platforms to ensure data collection can be use outdoors

Technical skills

- **Machine learning CV:** PyTorch, Yolov5, RetinaNet, ResNet, Python
- **IDE:** Google colab, PyCharm, Jupyter Notebooks, Eclipse, IntelliJ Idea
- **Web development:** JavaScript (vue.js, uni-app), spring boot, bootstrap
- **Others:** Linux, AWS, Unity, VR/AR, Java, JavaScript, C++, Docker, Git