

Wenkai Ren

Email: wren2@jhu.edu
Phone: (518)9154181
Website: www.github.com/wenkairan

EDUCATION

Johns Hopkins University , Baltimore, MD	Aug.2018 - Dec.2019
M.S. in Robotics Engineering (Laboratory Computational Sensing and Robotics)	
Columbia University, Graduate school of Engineering and Applied Science , New York, NY	Aug.2016 - Dec.2017
M.S. in Mechanical Engineering	
Northern Arizona University , Flagstaff, AZ	Aug.2013 - May.2016
B.S. in Mechanical Engineering	
Minor in Electrical Engineering and Mathematics	
GPA: 3.75/4.0 (cum laude)	

EXPERIENCE

3D Food Printing , Columbia University, NY	May.2016 - Oct.2017
Research Assistant , Supervisor: Hod Lipson	
• Programmed for food print including switching materials, controlling layer height and visualizing the process with python	
• Implemented the food printer with laser sensor (1%) for high-accuracy feedback control	
Association Unmanned Vehicle System International Competition (Undergraduate Capstone Design, Team)	Aug.2015 - May.2016
• Designed a fully autonomous underwater robot can perform color identification, dropping markers, and gripping system	
• My work includes motor brackets and water sealing end caps design, front camera vision detection based on color marker and gripping control system.	

PROJECT

3D Perception based on RGB-D Sensor , ROS	June.2018
• Implement PR2 robot in the simulator with RGB-D sensor data to pick up a listed objects in a given table and place the corresponding dropbox	
• Implement RANSAC plane segmentation and Euclidean Clustering to separate the objects in perception pipeline	
Search and Sample Return Project (modified after NASA sample return challenge), ROS	May.2018
• Implement the “rover” car in unit engine simulator to autonomously map a simulated environment and search for samples of interest	
• Use “hug wall” method in decision pipeline to map the whole environment and collecting the samples without stuck.	
Udacity Self-driving nano-degree Program (Python, C++)	Jun.2017 - Feb.2018
• System Integration Project (Team Leader)	
Use ROS nodes to implement core autonomous vehicle system based on the Udacity self-driving car “Carla” architecture with three module includes perception , control, and path planning.	
perception: based on camera image to perceive traffic light	
planning: based on the next waypoint and the decision of the traffic light waypoint to generate future waypoint	
control: based on the planning data of the trajectory, by sending control commends to steering, throttle and brake values.	
• Adopt PID control and model predictive control method for self-driving car in the simulator	
• Implemented extended and unscented Kalman Filter to track surrounding pedestrians and cars in the simulator with C++	
• Implemented Particle filter with Map Localization in vehicle simulated environment with C++	
• Trained an agent imitating human’s driving behavior in the simulator based on NVIDIA model with Keras	
• Detected and tracked vehicles using sliding window methods with SVM classifier	
• Trained a convolutional neural network to classify traffic sign with 96% test accuracy with Tensorflow	
Drone follower , Tensorflow, keras	March.2018
• Built and trained an FCN to find a specific person in images from a simulated quad-copter	
Reinforcement Learning in Quadrator Navigation , Group Course Project	Oct.2017 - Dec.2017
• Built simulation environment with Air Sim using Unreal engine	
• Implemented the deep Q-learning, double Q-learning method for test	
Robot Arm Motion Planning , Course Project, ROS	Nov.2017
• Based on Kuka 7 joint arm, simulate cartesian controller to control the robot arm’s work space	
• Implement robot ending-factor to reach desired pose with collision free by rapid-exploring random tree(RRT)	
Raspberry Pi Programming for GoPiGo Robot , Course Project	Dec.2016
• Implemented “Bug2” algorithm on Go robot along with the ultrasound sensor data.	
• Use Picamera API to constantly taking photos and enable robot to recognize markers on photos instantly on raspberry Pi	

LANGUAGES & SKILLS

- *Python, C++/C, Java, R*
- *ROS, MATLAB, OpenCv, Tensorflow, Keras, Linux, git, Solidworks*