Dingding Zheng

(+1)445-888-1792 | zddkj@alumni.upenn.edu | Homepage | Github

Education

University of Pennsylvania — Philadelphia, USA
Master of Science in Engineering, Robotics — *Aug. 2018 – May 2020*

Donghua University — Shanghai, CHINA
Bachelor of Science in Engineering, Electrical Engineering — *Sep. 2014 – June 2018*

Research Interests

Micro Aerial Vehicles, Multi-agent Systems, Human Robot Teaming, Visual-SLAM, Optimal Control

Research Experiences

Distributed and Collaborative Intelligent Systems and Technology — July 2020 – Present
University of Pennsylvania, Kumar lab — *Supervisor: Prof. Vijay Kumar*

- Researching on a variant of the multi-player reach-avoid game played between intruders and defenders with applications to **Perimeter-Defense**. Currently developing Perimeter-Defense algorithms for 30 Warthog robots.
- Implemented behavior cloning and task allocation algorithms for multi-robot system using Graph Neural Network.
- Improved the ROS, Unity-based Docker testing environment for heterogeneous systems.

Reconfiguration of Multi-modular Robots (SMOREs) — Aug 2019 – July 2020
University of Pennsylvania, Modlab — *Supervisor: Prof. Mark Yim*

- Researched on decreasing 3D reconfiguration empirical run-time of modular robots.
- Initialized multi-camera detection system using "Vicon" and "Apriltag".
- Implemented a MPC controller using ROS tuw package to do multi-robot path-planing and tracking.
- Explored new ways to justify the similarity between different robot configurations (topology).

Human Robot Interaction & Safety Guarantee for Multi-agent System — Aug 2019 – Oct 2019
University of Pennsylvania, PRECISE lab — *Supervisor: Prof. Osbert Bastani*

- Added car dynamics model into OpenAI multi-agent particle environment.
- Used MPC controller to simulate the human decision making and trained robotcar using MADDPG algorithm.
- Used "Human Social Force Model" to simulate human decision making and MPC controller to control robot car.

Teaching Experiences

MEAM 620 (Advanced Robotics) Teaching Assistant — Spring 2021
University of Pennsylvania, School of Engineering and Applied Science

Projects

Quadrotor Planning and Control | *Python, ROS, CrazyFlies 2.0, EuRoc, Vicon* — Feb 2020 – May 2020

- Designed a geometric non-linear PID controller to let the quadrotor reach its desired goal without collision.
- Down-sampled the path derived from A* to get waypoints and implemented "Minimum Jerk" algorithm to get optimal trajectory for quadrotor to track.
- Added corridor constraints on keyframes to generate aggressive but safe trajectories.
- Implemented "Complementary Filter" and RANSAC to get accurate estimated states.
- Estimated the pose of quadrotor given by data from IMU, onboard stereo pair using "Error-state Kalman Filter".
- Reconstructed a 3D environment model given data from EuRoc dataset.

Autonomous Racing | *ROS, C++, Python, F1tenth racing car* — Feb 2020 – May 2020

- Designed and implemented algorithms to let the "F1tenth" racing car finish loops as soon as possible. Ranked **1st** among all racing teams.
- Implemented "Point-to-Line Iterative Closest Point (PLICP)" to estimate the pose of racing car given by data from IMU, Lidar and VESC.
- Generated a 2D map of Upenn Levine 2nd floor using "Google Cartographer".

- Used "RRT*" algorithm and created local occupancy map to let the car avoid obstacles more efficiently.
- Implemented "Minimum Curvature" and "Covariance Matrix Adaptation Evolution Strat-egy (CMA-ES)" algorithm to get optimal racing trajectory.
- Designed "Obstacle-Dependent Gaussian Potential Field" algorithm to do obstacle avoidance.

RGBD SLAM | *ROS, Python, THOR-OP humanoid robot* Apr 2020 – May 2020

- Integrated the IMU orientation and odometry information from a walking humanoid with a 2D laser range scanner in order to build a 2D occupancy grid map of the walls and obstacles in the environment.
- Built a textured map by integrating additional camera and depth imagery from a Kinect One sensor.

Deep Learning for Computer Vision | *PyTorch, OpenAI gym* Aug 2019 – Dec 2019

- Generated adversarial images using deep neural network.
- Implemented "YOLO v1" to do extremely fast real time multi object detection.
- Implemented "Mask-RCNN", which combines object detection and semantic segementation.
- Implemented a family of generative models including: "Variational Autoencoder (VAE)"and "Generative Adversarial Network (GAN)".
- Controlled OpenAI racing-car v0 using "Clipped Proximal Policy Optimization (Clipped PPO)" algorithm. **Ranked (5/26)th** in the final race.

Orienation Tracking based Panorama Stitching | *Python* Apr 2019 – May 2019

- Implemented a kalman filter to track three dimensional orientation.
- Given IMU sensor readings from gyroscopes and accelerometers, estimated the underlying 3D orientation by learning the appropriate model parameters from ground truth data given by a Vicon motion capture system.
- Generated real-time panoramic images from camera images using the 3D orientation filter.

6-DoF Pose Estimation of an Oilcan | *PyTorch* Apr 2020 – May 2019

- Trained a heatmap-based neural network which estimates the location of the keypoints in the oilcan image.
- Synthesized the heatmaps by identifying the location of each keypoint on the 2D image and placing a 2D Gaussian centered on this location on the corresponding heatmap.
- Used the coordinates of detected keypoints to estimate the 6-DoF pose of the object (oilcan).

Barrel Detection using Color Segmentation based on GMMs | *Python* Mar 2019 – Apr 2019

- Trained a GMM-based model to detect barrels in images and found the relative world coordinates of the barrel.
- Implemented algorithms to learn the color model, segment the target color and finally localize the target object.
- Hand-labeled the training sets and then built a color classifier and a red barrel detector.

UPenn Engineering Logo Projection | *Matlab* Mar 2019 – Apr 2019

- Estimated the homography that maps the video images onto the logo points.
- Warped the sampled points according to the homography.
- Used the correspondence between sampled points and homography to project the "Penn Engineering" logo to the goal in a football match.

Scale Invariant Detection | *Matlab* Mar 2019 – Apr 2019

- Approximated a Laplacian of Gaussian filter (LoG) by a Difference of Gaussians (DoG).
- Used LoG filter for blob detection, such as: sunflower, birds, balloons, etc.

Awards & Activities

Excellent Academic Performance (Top 10%) | *Donghua University* 2016 - 2017
Excellent Volunteering | *1st International University Climbing Champion* 2017
Excellent Volunteering | *Shanghai International Marathon* 2015 - 2017
Mathematics Modeling Contest (Rank: 7/102) | *Donghua University* 2016
Mathematics Contest for Calculus (Rank: 3/351) | *Donghua University* 2015
Outstanding Undergraduate Student (Top 3%) | *Donghua University* 2015

Technical Skills

Languages: Python, C/C++, Matlab, HTML/CSS
Deep Learning Frameworks: Pytorch, Tensorflow, Keras
Developer Tools: Git, Docker, VS Code, Visual Studio, PyCharm
Libraries: Cvxopt, pandas, NumPy, OpenCV, scipy, pygame, Matplotlib