

Ran Cheng

An Intelligent Robotics Researcher

MLL 411, McGill University, Montreal, Canada

✉ ran.cheng2@mail.mcgill.ca

🏠 rancheng.github.io

📞 774-823-4304



EDUCATION

McGill University, Computer Science

Master of Science in Computer Science

Montreal, Canada

Expected May 2020

Thesis: Guided Robust Visual Navigation with Deep Learning

Relevant Coursework: Intelligent Robotics, Reinforcement Learning, Applied Machine Learning

Coursera

Online

Honored Degrees & Long-term Community Contributor

Aug 2015 - Aug 2017

Completed Courses: Neural Network for Machine Learning (UToronto), Robotics: Specialization (UPenn), Machine Learning (Stanford)

Tongji University, School of Software Engineering

Shanghai, China

Bachelor of Engineer, Software Engineering; GPA: 3.89/4.0

Aug 2011 - Aug 2015

Honors and Awards: Outstanding Diploma thesis, National Aspiration Fellowship, Second Class Prize Fellowship, Social Activism Award, IBM Outstanding Contribution Award, Microsoft Imagine Cup, FTC (First Tech Challenge, a Robot Competition Conference) Technician

EXPERIENCE

Mobile Robotics Lab, McGill University

Montreal, Canada

Research Assistant, Supervisor: **Gregory Dudek**

Sep 2017 - Now

- **Deep Monocular VO:** Designed a semi-supervised monocular depth estimator for video using sparse bundle adjustment in a sliding window, achieved 0.117 (top5) RMSE in **NYU v2** and 2.981 (top10) RMSE in **Kitti eigen-split**. backbone is Unet+CSPN, trained with semi-dense map point tracked by VO, implemented with PyTorch.
- **NavGuideNet:** A **synthesized hierarchical neural network** for autonomous navigation in complex environment and variant landscapes (tested in field/underwater environments). Backbone encoder is **Resnet18**, **latent code** was concatenated with control signals and decoder is **de-convolution network** (transposed convolution), average lost rate in underwater experiments decreased 91.5%, average aligned trajectory RMSE is 6.39 with improvement of 51.8% ([gitlab](#))
- **Deep RL Auto Driving (Sim2Real):** Re-implemented CAD2RL in python and extended to multiple policy gradient based backends (A3C+LSTM, DDPG, TRPO, PPO, TD3) in continuous action space, simulator is **Microsoft AirSim** tested on RC car and UAV

iLab Tongji/University of South California

Shanghai, China, Los Angeles, USA

Research Assistant, Supervisor: **Jianwei Lu, Laurent Itti**

Apr 2015 - Jul 2017

- **SLAM Fusion:** Vision (monocular) LiDar fusion with direct method (jointly optimize optical flow with Sparse Bundle Adjustment on ORB features) extra constraint from LiDar helps eliminating depth from null space.
- **Visual SLAM with Saliency:** joint optimizing the graph (G2O) with salient voting as extra binary edges.

UCLA

Los Angeles, USA

Part-time Research Assistant, Supervisor: **Yi Xing**

Jul 2015 - Jan 2016

- **Code Parallelization:** optimized their RNA analysis tool, [stable release (rMATS 3.0.9)], binding the large matrix calculations with C11 (SSE/AVX vectorization, Intel) and CUDA (cuBLAS, Nvidia)

PROJECTS

- **Visual SLAM:** Refactored **Stereo-DSO** ([gitlab](#)) with keyframe and feature point based **loop closure**, **BA** (Ceres solver). Using Tracking-Mapping-ReTracking three thread architecture, run on Nvidia-TX2 with 15 FPS. Comprehensively **re-implemented DSO** and annotated with exhaustive explains.
- **Deep Monocular Dense 3D Reconstruction:** Dense 3D reconstruction with **monodepth2** initialized Visual Odometry, leveraging traditional photometric consistency, occlusion discrepancy, and local geometrical-smooth assumptions to **optimize depth estimation** (LM method) and **register** 3D map point clouds.
- **Abstraction Augmented Deep RL:** Abstract rgb image with Unet shaped network to digest image in latent representation, and learn from latent inputs, average convergence time increased 27.3%, maximum reward (10M iterations) is 1.21 times than baseline model without abstraction augmentation, experiments conducted under self-collected dataset from AirSim simulator ([github](#))
- **Forgetting Model for BP:** Introduced forgetting model for back propagation as in gradient dynamic routine, inspired from forgetting curve, I invented forgetting factor to regulate delta weights updates, ([math proof](#))
- **LOAM:** extended LOAM with co-visibility check, optimized with Ceres optimizer and asynchronous threading

PUBLICATIONS

- Deep Monocular Visual Odometry with Direct Method, **R. Cheng**, D. Meger, G. Dudek **CVPR 2020** [preprint](#)
- Robust Off-road Robotics Visual Navigation with Back-Projection Learning, **R. Cheng**, G. Dudek, D. Meger, **ICRA 2020**, [preprint](#)
- Abstraction Augmentation for Deep Reinforcement Learning, **R. Cheng**, G. Dudek, D. Meger **CoRL 2019**, [preprint](#)
- Navigation in the Service of Enhanced Pose Estimation, Travis Manderson, **Ran Cheng**, David Meger and Gregory Dudek, **ISER 2018**, [paper](#)
- Vision-Based Autonomous Underwater Swimming in Dense Coral for Combined Collision Avoidance and Target Selection, T. Manderson, **R. Cheng**, D. Meger, G. Dudek, **IROS 2018**, [paper](#)