Xiaoxin Zhou

Zxx914490390[at]Gmail[dot]com | GitHub | LinkedIn | 6478352256 | Toronto, Canada

SUMMARY OF QUALIFICATIONS

- Skilled in the following programming languages: Python, MATLAB, BASH, and LaTeX.
- Proficiency in the programming languages: SQL, Smalltalk, Rust, Elixir and Prolog; Advanced level of OpenCV and Scikit learn in Python; Eigen in C++; In-depth knowledge of Linux command prompt.
- Strong analytical, problem-solving skills, and make well thought out decisions; Perform well under pressure for an extended period; Hands-on with many SLAM systems such as ORB SLAM, Kimera and experience in image reconstruction and Colmap (point cloud) research.

PROFESSIONAL EXPERIENCE

Robotics and Computer Vision Lab (RCVL) **Nov 2020 – Present**

Research Assistant, Ryerson University

- Actively cooperate with Dr. Sajad Saeedi to research a new efficient map representation for robotic applications.
- Study Visual SLAM, Visual-inertial SLAM, Visual Odometry, Ubuntu and ROS.
- Hands-on Eigen in C++, ORB SLAM2, Kimera, superGlue, GMAPPING on ROS and Gazebo; feature extraction use FAST, BRIEF, ORB, SIFT, Affine-SIFT, Affine-ORB.
- Use Colmap to generative point cloud from images and projection from 3D point cloud to 2D plane OpenCV on Python.

Data Mining **Jan 2021 – Apr 2022**

Ryerson University

- In depth knowledge basic data mining techniques, data preprocessing, association rule mining, classification, clustering, web mining, and data mining applications (e.g., web personalization, recommender system, security).

Machine Learning **Sep 2021 – Dec 2021**

Team Lead, Ryerson University

- Applied exploratory data analysis to investigate and eliminate non-essential data. Used Scikit-learn and Keras to set up models.
- Used Logistic Regression, Support Vector Classification, Random Forest Classifier and Neural Network to predict the Titanic survival rate. The final model has captured 80%+ validation in the Neural network (Python).

Computer Vision **Sep 2021 – Dec 2021**

Team Lead, Ryerson University

- Understood image formation process, image representation, feature extraction, model fitting, motion analysis, 3D parameter estimation and applications (MATLAB).
- Lead the group to reduce the number of keypoint for ASIFT by over 96% using QuadTree, keep the distribution for each keypoints, and find the best descriptor matching algorithm (Python).
- Hands-On Robert, Prewitt and Sobel edge detector, MLESAC, Low-Light Image Enhancement, Camera Calibration, structure from motion (MATLAB).

Autonomous Mobile Robotics **Sep 2021 – Dec 2021**

Ryerson University

- Run ELEGOO robot by direction control, edge detection and object detection. (Arduino).

EDUCATION

BSc (Hons), Computer Science **May 2022**

Ryerson University, Cum. GPA: 3.90/4.33

AWARD

Ryerson FEAS-URO **May 2021 – Aug 2021**

- Supported by the Ryerson University Faculty of Engineering and Architectural Science Undergraduate Research Opportunity, the value of $10,471.80.

PROJECTS/PAPERS

Undergraduate Thesis **Sep 2021 – Apr 2022**

- Determined the best feature matching performance from SIFT, Affine-SIFT, ORB, and CNN feature extraction methods. The result shows the CNN feature extraction methods have the most matching rate 24.213%.

Analysis Data by Association Analysis and Clustering Analysis **Jan 2021 – Apr 2022**

- Utilized Car Evaluation Data Set to solve several practical problems: association analysis and clustering analysis.

Titanic Machine Learning from Disaster **Sep 2021 – Dec 2021**

- Exploit four different models: Logistic Regression, Support Vector, Random Forest, and Neural Net to predict the Titanic survival rate. The result shows the Neural Net have the highest accuracy 80.28%

Apply QuadTree on ASIFT Reduce Keypoints **Sep 2021 – Dec 2021**

- Adopt the QuadTree method to reduce the SIFT and Affine-SIFT keypoints. Reduced over 95% keypoints from Affine-SIFT and increased the feature matching time.

Safe Driving Assist **Sep 2021 – Dec 2021**

- Develop a semi-autonomous control scheme that works well to keep the drivers safe while not annoying them. Implement the autonomous override logic and auto avoid obstacles, the user can use the joystick to control the robot's action wirelessly.