Zexi Han

+86-13120333161 | zexihan@outlook.com Shanghai
zexihan.com/portfolio | linkedin.com/in/zexihan | github.com/zexihan CN

PROFESSIONAL SKILLS

Programming Languages: C++, Python, Java, JavaScript, R, MATLAB
Computer Vision: 2D: CNN, YOLOs; 3D: Point-based / Voxel-based / Center-based Object Detection; Sensors: RGB-D, LiDAR
Libraries and Frameworks: PyTorch, OpenCV, Eigen, OpenMP, PCL, MapReduce, Linux, AWS, Pandas, scikit-learn, Docker, Git

EDUCATION

Northeastern University, Boston, MA, USA Jan 2017 – May 2019
*Master of Science in **Data Science*** GPA 3.8 / 4

Beijing University of Posts and Telecommunications, Beijing, China Sept 2012 – Jun 2016
Joint Program with **Queen Mary University of London** GPA 3.5 / 4
*Bachelor of Science in **Telecommunications Engineering** with the First Class Honors*

- Awards: Outstanding Final Thesis (Rank top 2%)

PROFESSIONAL EXPERIENCE

Software Engineer, Inceptio Technology | Perception – Autonomous Driving, Shanghai Oct 2020 – Present

Project: LiDAR Perception for Autonomous Driving Trucks | Area: Computer Vision, 3D Deep Learning

- Responsible for the development of the real-time multi-task LiDAR perception deep learning algorithms for the L3/L4 autonomous driving trucks, including object detection, tracking, velocity estimation, freespace, and lane detection.
- Deploy the model to SoC by optimizing the inference time-efficiency of the model from multiple perspectives.
- Coordinate with other AD teams to push the L3 trucks to mass production in a limited time.

Software Engineer, Suning Commerce R&D Center | Applied AI Lab, Palo Alto, CA Aug 2019 – Aug 2020

Project: Point Cloud Pedestrian Detection | Area: Computer Vision, 3D Deep Learning

- Led applied research of the point cloud pedestrian detection and tracking algorithms using RGB-D sensors for cashier-less automated convenience stores; the model was optimized to reach an average precision of 0.93.
- Built a large-scale 3D pedestrian detection dataset in an iterative and evolutive annotation process. Developed a point cloud key point annotation tool with three.js, and a bird's eye view rotated bounding box annotation tool based on labelImg in PyQt.
- Optimized the speed of the associated real-time point cloud preprocessing/postprocessing from seconds to milliseconds in C++.
- Trained PointRCNN / Part-A^2 / PV-RCNN models for 3D pedestrian detection, which are optimized on adding optical flow attention, instance augmentation, and tuning parameters including learning rate, anchor size, epochs, and non-max suppression.

Project: Video Data Mining and Content Understanding | Area: Computer Vision

- Achieved automatic short video tagging by mining image cues from OCR, image classification, and face recognition modules.

Data Scientist Co-op, Rue Gilt Groupe, Boston, MA Jan 2018 – Jun 2018

Project: Reseller Identification | Area: Machine Learning, Data Mining and Engineering

- Worked on feature engineering and XGBoost model training from an iterative perspective to identify resellers from over 2 million buyers, which was put into production to provide resellers with personalized boutique recommendations.
- Built docker apps for the pipeline of feature extraction, model training, and inference, and deployed to Amazon ECS and Airflow.
- Maintained daily ETL process for the recommendation system with robust SQL on Snowflake.

Research Assistant, CASIA | National Laboratory of Pattern Recognition, Beijing Aug 2015 – May 2016

Project: Image Retrieval | Area: Computer vision, Deep Learning

- Involved in designing and building a Three-stage Hybrid Visual Search Framework (Classification, Object Detection and Matching) to the task of same-sku commercial product image retrieval with convolutional neural networks.
- Trained classification models with multiple CNN backbones on Taobao 5M commercial product images using Caffe.
- Extracted object semantic features by finetuning class-specific Faster R-CNN models for object localization.
- Achieved real-time image retrieval performance (552.49ms) with model compression and developed Android demo mobile application.

PROJECT EXPERIENCE

Self-Driving Car System Integration, Udacity Jan 2020 – May 2020

- Integrated perception, path planning and control modules in ROS to maneuver a simulated autonomous vehicle on road while being able to stop at red traffic lights, switch lanes, and safely overtake vehicles in front.
- Use both traditional and deep learning algorithms for vehicle perception, including lane boundary identification and traffic sign recognition.
- Applied Extended Kalman Filter in C++ for sensor fusion to predict locations of other vehicles with certainty.

TripElf – Interactive-Map Web App with Neighborhood-Level Airbnb Review Summarization, NU Jan 2019 – Apr 2019

- Proposed and developed an application to help travelers pick their favorite short-term rental neighborhoods before traveling by demonstrating the machine-generated overviews of the neighborhoods.
- Implemented an interactive map frontend with React and Mapbox GL JS for demonstration of neighborhood profiles.
- Experimented to optimize the scalability by horizontal data sharding with consistent hashing for the Cassandra database.
- Applied text models, such as KL-Sum, LDA-Sum and ELMo, to summarize Airbnb reviews and generate neighborhood overview of different aspects from travelers' point of view, including entertainment, noise, safety, transit, expense, and host review.

TuneS – Social Music Website Jan 2019 – Apr 2019

- Developed a single page application using MERN stack and Spotify Web API that serves for music fans to engage with other music lovers and discover new songs, albums, and artists.
- Handled OAuth authorization, like/share/follow functions with a RESTful API built in Express and MongoDB backend.
- Improved database concurrency performance by integrating Redis as cache for counters, music charts, and other services.
- Designed and developed a responsive and interactive React frontend utilizing Bootstrap and AJAX techniques.

Parallel Matrix Multiplication in MapReduce, NU Oct 2018 – Dec 2018

- Implemented parallelization mechanisms for large matrix multiplication in MapReduce in distributed settings, including Horizontal-Vertical Partitioning and Vertical-Horizontal Partitioning for synthetic dense and sparse matrices.
- Measured speedup and scalability performance for the two intelligent partitioning methods on Amazon EMR and S3.