|  |  |
| --- | --- |
| **WORK EXPERIENCE** |  |
| **Momenta** Software Engineer Intern | Shanghai, China | ***Feb 2024 – June 2024*** |
| *Python/Algorithm Development/Data Processing/Clustering/Git*   * Developed a **clustering algorithm** to detect stuck states in autonomous vehicles, boosting recovery performance of GM Cadillac Lyric by **3%** across **800+** **real** parking test cases in **30+** garages within **4 months** * Engineered a robust checker to detect prolonged braking stops, achieving **98%** accuracy in identifying stuck states and reducing false positives by **15%**, enhancing simulation reliability across **15,000+ events** * Partnered closely with the product manager and designed a **Python class library** to automatically process **36,000+** simulation reports across **6** parking scenarios, improving analysis efficiency by **87.5%** | |
| **TECHNICAL LEADERSHIP** |  |
| **Cal Hackthon 11.0 | A LLM-Powered Drive-Thru Solution** | Team Leader | ***Sept 2024 - Nov 2024*** |
| *Deepgram/AutoGen/Flask/Vue.js/Fine-tuning*   * Designed a **sequential chat** system with 3 LLM agents using **AutoGen** to analyze user requirements and generate ordered items, achieving **0.9551** cosine similarity, **0.2712** ROUGE-L, and **0.8811** BERT F1 score. * Developed data processing functions and effective prompts for LLM agents, and integrated **Deepgram API** in a **Flask** backend to convert speech to text for **real-time** drive-thru interaction, achieving **85%** successful transactions | |
| **DJI RoboMaster Competition (4 Years)** | Team Leader & Computer Vision Engineer | ***Sept 2020 - June 2024*** |
| *C++/OpenCV/YOLOv7/Real-time system/Linux/Least squares* | **[Git repo](https://github.com/SRM-Vision/SRM-Vision-2022)** | **[Video](https://www.youtube.com/watch?v=4uyBBJRXUTg)**   * Led a **40-student** team to build **8 types** of robots from scratch to product, winning the **3rd place in RoboMaster 2023** * Developed a **real-time** **auto-aim system** with monocular camera input for **mobile robots** on NVIDIA NX, achieving **60 fps** with over **90%** accuracy in **C++/Linux** using **OpenCV** and **YOLOv7** for object detection * Implemented a **trajectory prediction** algorithm using a **least squares** algorithm, improving system efficiency by **50%** | |
| **FIRST Tech Challenge (3 Years)** | Team Leader, 14263/16107 F.G.(Facing The Giants) | ***Sept 2017 - Jan 2020*** |
| *TensorFlow/OpenCV/Leadership/Motor control/Rule-based strategy* | **[Team Documentary](https://www.youtube.com/watch?v=ZKn0rDUpNfY)**   * Developed an autonomous system using motor encoders, color sensors, and **OpenCV**/TensorFlow SDK for control and **95%+** accurate detection, achieving **highest** score in the Regional with **rule-based** human driver imitating strategies * Led the team to achieve 2 admissions into FIRST World Championships in [2018](https://theorangealliance.org/teams/14263?season_key=1718)&[2019](https://theorangealliance.org/teams/16107?season_key=1819)(**top 2 %** out of 7500 teams globally), 1 Inspire Award(**1st** out of 40 teams), and 3 Connect Awards**(top 8%** out of 60 teams) | |
| **RESEARCH** |  |
| **Visual Explainer For Deep Learning Decisions** | Research Assistant | ***Sept 2023 – May 2024*** |
| *Full stack/Python/PyTorch/AutoEncoder/Semantic Segmentation/Django/Vue.js* | **[Demo Video](https://www.youtube.com/watch?si=gGqRFG9EwhzPH2b1&v=HcAEPgrM9zM&feature=youtu.be)**   * Designed a 2-stage **semantic segmentation** and an **AutoEncoder** with tree constraints to extract and rank concepts by importance using Shapley Value, boosting consistency score by **35%** on **1000+** images from **20 ImageNet classes** * Developed a **Django backend APIs** for page navigation, handling **GET** and **POST** requests, and efficient data retrieval * Built a **Vue.js** **frontend** showing features such as user login, image segmentation, and contribution heatmap visualization | |
| **Mining Property Relations of NASICON Solid Electrolyte** | Research Assistant | ***Sept 2021 - May 2023*** |
| *Full stack/Python/Java/Vue.js/SpringBoot/Py2Neo/Neo4j/BERT/Element UI*   * Labeled **7,000+** high-quality NASICON literature sentences, improving Named Entity Recognition (NER) model performance by **5%** in precision, **3%** in recall, and **4%** in F-1 score * Developed a **BERT**-based data processing pipeline to extract **106,896** material entities and **260,475** entity-relation triples from **1,808** NASICON-related literature sources, with efficient storage in **Neo4j** and **MySQL** as backend database * Built a **Vue.js** platform with **Element UI**, **routing**, and **state management**, allowing materials scientists to identify target texts in literature and convert them into a **knowledge graph** to explore relationships between material properties | |
| **SKILLS & AFFINITIES** | |
| **Programming & Framework**: Python, C++, PyTorch, ROS, SQL, Linux, Vue.js, Django  **Library & Tools:** OpenCV, AutoGen, Deepgram, Py2Neo, Timor-Python, Transformers, Git, Neo4j, AWS, Galileo AI, Figma  **Affinities**: **DJI** *Event Tech Support Lead* (Apr 2024) **| FIRST** *Lead Robot Inspector* (Jan 2024), *Robot Inspector* (Mar 2021) | |