

# XIAO FAN

90 Fawcett St, Unit 114, Cambridge MA 02138

☎ 626-567-1840

✉ [xffan@caltech.edu](mailto:xffan@caltech.edu)

🌐 [linkedin.com/in/xiao-fan-6a21b7171](https://www.linkedin.com/in/xiao-fan-6a21b7171)

🐙 [github.com/tommyfan34](https://github.com/tommyfan34)

## Education

### California Institute of Technology (Caltech)

Sep. 2021 – Dec. 2022 (Expected)

*Master of Science in Electrical and Electronics Engineering* | GPA: 4.1 / 4.3

*Pasadena, CA, US*

- **Core courses:** Network Structure; Software Engineering; Relational Databases; Web Development; Operating System; Programming Languages; Machine Learning & Data Mining; Functional Programming; Distributed Computing

### Zhejiang University

Sep. 2016 – Jul. 2020

*Bachelor of Engineering in Electronic Information Engineering* | GPA: 3.92 / 4.0 (top 3%)

*Hangzhou, China*

- **Core courses:** Data Structure; Computer Networks; C Programming; Linear Algebra; Calculus; Probability and Statistics

## Technical Skills

**Languages:** Java, Python, JavaScript, C#, C, Haskell, OCaml, Rust, Matlab/Simulink, HTML/CSS, SQL,  $\text{\LaTeX}$

**Developer Tools:** VS Code, IntelliJ IDEA, Android Studio, Unity 3D, Unreal Engine, Git, SVN, Docker

**Frameworks:** Express, Flask, Mongoose, Bootstrap, Java Spark, .NET

## Experience

### United Imaging Intelligence America

Jun. 2022 – Present

*Software Engineer Intern*

*Cambridge, MA*

- Developed an VR **Unity** app from scratch which synchronizes the robot in the virtual and the real world scene and visualizes point clouds as well as video streams. Used **Docker** for release delivery.
- Developed a RESTful DLL SDK for bridging the Unity app and the remote robot via HTTP using **.NET Standard**.
- Integrated deep learning results including body mesh estimation, body pose detection and RGB-D camera calibration into the Unity app.

### JOYNEXT Inc.

Jul. 2020 – Dec. 2020

*Software Engineer Intern*

*China*

- Conducted Vehicle-to-Everything (V2X) and Driver Monitor System research. **3 fields related patents granted**.
- Deployed botframework service at Azure using **C#** to integrate a virtual assistant into In-Vehicle Infotainment (IVI) System; Implemented an **Android client** at IVI which realizes Speech-To-Text and intelligent responses.
- Implemented an interactive animated avatar rendered in browser using **three.js**; Integrated **face-api.js** to recognize user face and rotate the avatar accordingly.
- Developed a **Simulink-Unreal Engine** co-simulate platform to compute and visualize V2X scenarios.
- Initiated and maintained a patent application management system which utilized the combination of **SVN**, **Python** and **Excel VBA**, supporting 200+ internal users.

## Projects

### Spreadsheet Engine | Python

Jan. 2022 - Mar. 2022

- Implemented a spreadsheet back-end engine which mimics the Google Sheets using **Python**.
- Added support for functions in spreadsheet formulas using **Lark** Json parser.
- Applied Tarjan algorithm to detect strongly connected components in the cell reference graph for cycle detection.
- Followed test-driven develop principle. Used **GitHub Action** for CI/CD, **coverage.py** and **cProfile** for code quality and performance profiling, as well as **Pylint** for code style analyze. Package management with **pipenv**.

### Yelpcamp | Bootstrap, Express, Node.js, MongoDB

Aug. 2021

- Set up a campground review website from scratch. Front-end built with **Bootstrap** and **EJS**, supporting responsive display. Utilized **Node.js** and **Express** as back-end server framework.
- Employed **MongoDB** as no-SQL database to store user and session data. Connected with app using **Mongoose**.
- Deployed the database and app with **MongoDB atlas** and **Heroku**, respectively.

### BearMap | Java Spark, Apache Maven

Apr. 2021

- Implemented a map web app that supports map displaying, zooming, routing and autocompletion. Backend built with **Java Spark** framework, converting the HTML request to corresponding map rasters and visualize at front end.
- Constructed a **SAX XML parser** to format and build a graph representation