Name: Zihan Zhao
Tel: +1 5107357280, **email**: zihanzhao1117@berkeley.edu,
Web: https://hanzhao01.github.io/ **LinkedIn**: https://www.linkedin.com/in/zihan-zhao-466a7b223/

EDUCATION

AUG 2021 – MAY 2025: Computer Science (Bachelor's degree) at UC Berkeley

GPA: 3.870 / 4.000

Relevant Coursework:

- **CS:** The Structure and Interpretation of Computer Programs, Data Structures, Algorithm and Interactable Problems, Computer Architecture, Operating System and System Programming, Internet Architecture, Database Systems, Machine Learning, Full-Stack Web Development
- **Math:** Multivariable Calculus, Linear Algebra, Discrete Mathematics and Probability Theory, Differential Equations and Circuit Design

SKILLS

- **Languages**: Java, Python, C, Rust, Go, Node.js, JavaScript, HTML, CSS, SQL, Assembly, Scheme
- **Frameworks:** Express.js, React, Django, Gin
- **Development Tools:** Git, Jupyter Notebook, Logisim
- **Machine Learning Libraries:** PyTorch, Sklearn, Numpy, Pandas, Matplotlib

EXPERIENCE & PROJECTS:

Software Engineering Intern at Tencent - Monitoring System Bypass Configuration **May 2024 - August 2024**

・Designed and implemented a monitoring system bypass configuration, addressing the issue of high load and latency caused by the traditional backend method of fetching data for individual projects. When creating a project, the system determines whether to enable bypass, allowing projects with bypass enabled to periodically batch fetch user performance data and logs from external monitoring systems after going live. The data is written into memory and synchronized to the data warehouse, significantly improving data processing efficiency and reducing the load on the main system. The system can also automatically initialize bypass configuration upon startup.

Software Engineering Intern at Tencent - Low-Code Development Platform AI Assistance **May 2024 - August 2024**

・The AI system of the low-code development platform is used to check process node configurations, overall process logic, and embedded code modules, generating reports and alerts to help developers identify configuration and logic errors. My responsibilities included: preprocessing code modules, extracting functions, variables, and multi-level dependencies to optimize code checks; engineering verification of SQL and table creation statements, using EXPLAIN SQL to check database node configurations; displaying and allowing internal management to track the review status of process inspection reports in the backend management system, supporting fuzzy search, and optimizing data loading speed through pagination and multithreading.

EduHive - Educational Technology Platform **Feb 2024 – Present**

・EduHive is an educational technology platform I co-founded in 2024 with several other founders, aimed at connecting students, researchers, and entrepreneurs globally. The platform uses an AI-driven project matching system to support collaboration in research and entrepreneurship, promoting cross-institution and interdisciplinary cooperation. I led the development of the recommendation system, managed frontend and backend architecture, and designed AI recommendation algorithms to help professors and students find project matches.

2D Third-Person Game **Apr 2023 - May 2023**

・Crafted a 2D game with dynamic map generation using Minimum Spanning Trees and Seed, employing stdDraw for rendering. Integrated a responsive engine for character movement, real-time status display, and customized character options with the ability to exit and save game progress.

RISC-V CPU Architecture Design **Oct 2023 - Nov 2023**

・Engineered a RISC-V CPU architecture with the development tool Logisim, integrating a full data path and key components for diverse instruction types, optimized with a two-stage pipeline for enhanced performance.

Implementation of Database System **Feb 2024 – May 2024**

・Constructed B+ Tree indexing, optimized join operations, applied join plan cost-based optimization with dynamic programming algorithm, and engineered transaction-safe concurrency operations, utilized the ARIES recovery algorithm to maintain the atomicity and durability of the transactions.

Convolutional Neural Network Implementations and Applications **March 2024 – April 2024**

・Implemented and tested key components of neural networks, including Activation Functions, Convolutional, Pooling, Flattening, and Fully Connected Layers, Forward Pass, and Backpropagation. Explored Batch Processing, ReLU, SGD and Adam Optimizer, Dropout Layers, Normalizing, and Regularizations to enhance efficiency. Using libraries like Torch and NumPy to apply CNN to real-world datasets including MNIST, and CIFAR-10.

Optimization of Convolution Operations in CNNs through Parallel Processing with C Programming **Nov 2023 – Dec 2023**

・Engineered and optimized convolution processes within CNNs using C programming, incorporating SIMD (Single Instruction, Multiple Data), OpenMP multi-threading, and Open MPI distributed computing techniques. Achieved a speed-up of 49.15x, enhancing processing efficiency and scalability.