

Guanshujie Fu

8700 Küsnacht,
Switzerland

[Personal Web](#)
fuguan@student.ethz.ch

EDUCATIONS

ETH Zürich (ETHz) M.S. in Information Technology and Electrical Engineering-EEIT • Specialized in <i>Computers and Networks</i>	Zürich, Switzerland 2023.09 - Present
University of Illinois, Urbana Champaign (UIUC) B.S. in Computer Engineering-ECE (GPA: 3.85/4.00) • Graduated with <i>High Honors</i> • Dean's List (2020&2021&2022)	Illinois, USA 2019.09 - 2023.06
Zhejiang University (ZJU) B.E. in Computer Engineering-ECE (GPA: 3.97/4.00) • Graduated as <i>Outstanding Graduates of Zhejiang University</i> • ZJU-UIUC scholarship (2020&2021)	Hangzhou, China 2019.09 - 2023.06

SKILLS

Programming:

Proficient in *C/C++*, *Go*, *Python*, *Java*, *Assembly*, *SystemVerilog*
Intermediate in *CUDA-C*, *HTML*, *JavaScript*, *RTL*, *P4*, *Haskell*, *MATLAB*.
Knowledge of *Linux Kernel*, *Network*, *Distributed System*, *Frontend Design*

Frameworks/Tools: *Docker*, *Kubernetes*, *Redis*, *Altera Quartus*, *Vitis/Vivado*, *Git*, *CMake*

Cloud/Database: *AWS*, *Azure*, *Spark*, *Hadoop*, *DynamoDB*, *MySQL*, *Milvus Vector Database*

EXPERIENCES

Backend Engineer Intern Hangzhou Houqi Tech Co. Ltd <i>Kubernetes</i> , <i>Docker</i> , <i>Redis</i> , <i>Golang</i> , <i>C++</i> • Worked in the develop group, my duty mainly includes deploying high performance vector database <i>Milvus</i> on <i>Kubernetes</i> cluster, developing vector store/search/query APIs in <i>Golang</i> based on <i>Milvus</i> , and developing/maintaining <i>Kubernetes</i> <i>nodes</i> and <i>Docker</i> containers to provide low-latency vector operations as a microservice in a larger picture • Used <i>Redis</i> as intermediate storage for vector search results, and implemented ranking based on Euclidean Distance • Used <i>ffmpeg</i> and <i>Hikvision C++ SDK</i> to develop a <i>rtsp</i> video stream pulling/pushing scheme. The pulled stream data is decoded and converted into <i>OpenCV Mat</i> format to fetch into a self-developed face detection algorithm within <i>30ms</i>	2023.02 - 2023.05
Undergraduate Researcher UIUC <i>Edge Computing</i> , <i>Xilinx FPGA</i> , <i>High Level Synthesis</i> , <i>C++</i> , <i>P4</i> Advisor: Professor Nam Sung Kim • Worked in the F.A.S.T lab, I explored the application of <i>Samsung SmartSSD</i> and <i>Xilinx FPGA</i> on Edge Computing. • Implemented benchmark program to test bandwidth performance of <i>SmartSSD</i> , designed data encryption algorithm (Run Length Encoding and LZ77) using <i>HLS C++</i> in <i>SmartSSD</i> • Offloaded database filter operations to <i>SmartSSD</i> to provide database operations using <i>HLS C++</i> , and tested the performance to verify the potential application of <i>SmartSSD</i> in database operations at data center • Worked with a Ph.D. candidate, I implemented a <i>Vitis P4</i> module and deployed it into <i>Corundum</i> high-performance FPGA-based NIC. The module will receive incoming network packets to split based head data and perform simple process tasks	2022.02 - 2023.02
Research Assistantship National University of Singapore <i>DPU</i> , <i>TCP/IP</i> , <i>C++</i> Advisor: Professor Jialin Li • Designed and implemented network application on <i>NVIDIA BlueField-2 DPU</i> with <i>DOCA Flow</i> acceleration • We explored the potential of offloading the main TCP stacks from Host <i>CPU</i> to <i>DPU</i> based on <i>DOCA Flow</i> framework, proposed potential offloading scheme and made presentation to industry group	2022.08 - 2022.11

PROJECTS

Full-Stack Website Development [Repo] UIUC • Worked in a team to develop a full stack website for movie dataset collection and user bias recommendation • Used <i>HTML</i> and <i>JavaScript</i> to construct web pages, <i>Python</i> -based <i>Flask</i> to render web pages • Used <i>MySQL</i> to manage backend large-scale movie dataset and <i>Google Cloud</i> to deploy website	2022.06 - 2022.08
Unix-like OS Kernel Design [Repo] UIUC • Led a team to design and implement an OS kernel resembling Linux with basic and advanced features in <i>C</i> and <i>Assembly</i> • The kernel includes file system, virtual memory, process management & scheduling, interrupts & exceptions and etc. • Designed a high-resolution (60fps, 800*600 resolution) graphic user interface with standard VGA capable	2022.03 - 2022.05

CUDA-based Optimization on Convolutional Layer [\[Repo\]](#) | UIUC

2021.09 – 2021.12

- Parallelized the forward pass of convolution layer with *CUDA C/C++* to run it efficiently on GPU
- Used optimization methods including Matrix Unrolling, Kernel Fusion and Reduction
- Used NVIDIA Nsight Systems to analyze and optimize

FPGA-based Graphic Design [\[Repo\]](#) | UIUC

2021.09 - 2021.12

- Developed an *FPGA-based* version of video game using *SystemVerilog* along with SoC, capable of processing and outputting complex graphics (60fps, 640*480 resolution) to VGA in a high frame rate and enabling keyboard control
- Used a *NIOS II SOC* to run the software game loop FPGA and communicate to the graphics system through the *Avalon Bus*
- Designed a complex finite state machine (FSM) on the FPGA board to optimize user interaction and collision detection

Representation and Extraction of Diesel Engine Maintenance Knowledge Graph | ZJU

2020.06 - 2021.02

Advisor: Professor Hongwei Wang

- Designed a framework in *Python* to extract bidirectional relations through a novel combination of reports preprocessing, *BERT* model and *Bi-LSTM-CRF* model
- Enabled the framework to construct diesel engine maintenance knowledge graph based on data set collected from power plants, automatically extract key information from the unstructured text in maintenance reports, transfer the extracted results into a structured knowledge graph using *Neo4j*, and construct bidirectional relations in the graph using *Protégé*

PUBLICATION

Jin Y., **Fu G.**, Qian L., Liu H., Wang H. “Representation and Extraction of Diesel Engine Maintenance Knowledge Graph with Bidirectional Relations Based on BERT and Bi-LSTM-CRF Model”, in 2021 IEEE International Conference on e-Business Engineering (ICEBE 2021), pp 126-133, Nov. 2021. [\[Paper\]](#)