Po-Chi, Hsu

https://github.com/Tommy-Hsu

Email: hpc880214@gmail.com

Mobile: +886-912-129-381

EDUCATION

- **National Yang Ming Chiao Tung University** — Hsinchu, TW
  Master of Science in Computer Science; GPA: 4.18 — *Sep 2022 – Jun 2025*

- **National Yunlin University of Science and Technology** — Yunlin, TW
  Bachelor of Science in Computer Science; GPA: 3.65 — *Sep 2017 – Jun 2022*

EXPERIENCE

- **Mediatek** — ChuPei, TW
  Software Engineer Intern (Windows Wi-Fi Driver) — *July 2023 – Aug 2023*
  - **Driver Development - Efuse**: Implemented driver operations to read and write efuse, configuring and managing internal circuit parameters. Patched event handling mechanisms to ensure process stability and accuracy.
  - **Driver Development - Sniffer Mode**: Extracted information from RX descriptors. Incorporated Radiotap and PCAP headers for advanced packet analysis. Established robust packet receiving and transmission workflows in sniffer mode to aid in network monitoring and debugging.

THESIS

- **Feature Alignment and Compositional Token for Human Pose Estimation**: Video-based 2D human pose estimation faces challenges such as motion blur, self-occlusions, and body parts extending beyond image boundaries. Existing heatmap-based methods leverage temporal information but struggle to model structural dependencies between human joints, leading to discontinuous or unrealistic predictions. FACT-Pose integrate spatiotemporal feature alignment and compositional token representations to robustly capture motion cues and joint dependencies, effectively addressing challenges like motion blur and occlusions.

COURSES

- **Physical Design Automation**: Circuit partitioning using the Fiduccia–Mattheyses heuristic to balance module sizes and minimize interconnections. Chip floorplanning with simulated annealing and B*-trees for optimized area and wirelength. Standard cell layout through Euler path methods to maximize diffusion sharing and reduce total HPWL. Channel routing employing a dogleg routing algorithm to efficiently place tracks and vias.

- **Parallel Programming**: Used pseudo SIMD intrinsics to simulate array exponential calculations and array sum operations. Guiding automatic loop vectorization using AVX2-specific compiler options and code hints. Monte Carlo method to estimate Pi using Pthread. Generating the Mandelbrot set using std::thread and analyzing. Profiled application performance with gprof. Parallelized Page Rank algorithm, Direction-Optimizing Breadth-First Search algorithm using OpenMP. Utilized different Open MPI communication functions. Implemented and benchmarked CUDA image processing kernels. Optimizing image convolution using OpenCL. Converted a C++ ray tracing algorithm to CUDA to leverage GPU parallelism.

- **Software Testing**: Engineered tests using unittest with mock, spies, and exception handling. Developed web surfing automation scripts with Selenium. Implemented an LLVM Pass to trace function calls by injecting code at entry/exit points, tracking call depth via global variables, and formatting output with indentation to visualize call hierarchy. Analyzing Address Sanitizer and Valgrind across different memory error scenarios. American Fuzzy Lop for fuzz testing and Angr for symbolic execution.

- **Cloud Native Development**: Developed Flask user microservice with RESTful API and integrated MongoDB for data persistence. Implemented tests using Python unittest and Jest. Deployed and managed microservices with Docker Swarm.

- **Network Programming**: ddd

- **Pattern Recognition**:

- **Image Processing**:

SKILLS

- **Languages**: C, C++, Python, CUDA
- **Tools**: GDB, Wireshark, Git, Docker, WinDbg, GitHub Actions, Valgrind, Address Sanitizer, LLVM
- **Frameworks**: PyTorch
- **Operating Systems**: Linux, Unix, Socket, Shell, Makefile, Boost