Peter Kim

pekim@umich.edu | pjk1m.com | github.com/PJ1229 | linkedin.com/in/pjk1m

Education

University of Michigan College of Engineering *Ann Arbor, MI*

Bachelor of Engineering in Electrical Engineering, Minor in Computer Science *Expected Graduation: May 2029*

Coursework: EECS 215* (Circuits), EECS 230* (Electromagnetics I), EECS 270 (Digital Logic), ROB 380* (Autonomous Robotics), EECS 281 (Data Structures & Algorithms), EECS 280 (Programming & Data Structures), EECS 203 (Discrete Math) *(*in progress)*

Wayne State University *Detroit, MI*

Completed coursework toward Bachelor of Engineering in Electrical Engineering *Aug 2024 – May 2025*

Research: Implemented and evaluated PyTorch denoising pipelines for AI diffusion models under faculty research in probabilistic generative models

Experience

Software Engineer Intern (Incoming) *Summer 2027*

Boston Scientific *Maple Grove, MN*

Software Engineering Co-Op / Intern *Aug 2025 – Present; May 2025 – Aug 2025*

Yazaki North America *Canton, MI*

- Built a wire-harness schematic QA pipeline (Python, pdfplumber/Tesseract OCR + validation rules) that auto-locates taping and layer defects in production drawings and exports annotated review bundles; root-caused a systematic defect driving ∼$44K savings per frame harness and cut review time 88% (128 hrs → <16 hrs)
- Owned an internal Python service end to end — design through deployment and support — extracting Assembly BOMs from Tesla EV wiring-harness data (Capital HDSI, KBL, PDF) via format-specific parsers with a vision-model OCR fallback; validated 17/17 parts on a production harness and 44/50 (88%) on image-only drawings vs. a 0-line incumbent baseline

Creative Video Intern *Aug 2026 – Present*

University of Michigan Athletics *Ann Arbor, MI*

- Built *um-autolog*, an on-device perception pipeline replacing hours of hand-typed game-footage logging: person detection localizes subjects, OCR is restricted to per-person torso crops, and candidates are ranked by salience-weighted voting across frames; encoded 29 sports and 269 action tags into a naming engine covered by 33 unit tests

Projects

Autonomous Gimbal Robot (In Progress) | *ROS 2 Jazzy, Raspberry Pi 5, Embedded Linux, LiDAR, C++, Python*

- Designed the electromechanical stack for a self-propelled omni-directional camera robot: sized and regulated a LiPo power system (load budget, rail regulation, current headroom), selected motor drivers off datasheet stall current for a holonomic drivetrain, and specified gimbal actuation and a laser-cut chassis
- Brought up an embedded Linux compute stack on Raspberry Pi 5 (Ubuntu 24.04, ROS 2 Jazzy) and integrated dual MIPI CSI cameras and a 2D LiDAR, writing ROS 2 nodes in C++/Python against vendor drivers and datasheets

Autonomous Snowplow Robot (Team SnowBlo) | *Arduino, Raspberry Pi, C/C++, Python, OpenCV, MQTT*

- Designed and built a self-propelled autonomous ground vehicle with a 4-person team, integrating a Raspberry Pi perception stack, Arduino motor-control firmware in C/C++, and a custom 3D-printed chassis into one working system
- Debugged hardware and software together to close the loop from sensing to actuation: AprilTag detection and OpenCV homography for bird's-eye pose estimation feeding a differential-drive navigation controller, with field-calibrated camera intrinsics and steering parameters

Technical Skills

Languages: C/C++, Python, MATLAB, Java, JavaScript, Excel VBA
Robotics & Perception: ROS 2 Jazzy, LiDAR, OpenCV, AprilTag, camera calibration, sensor integration, MQTT
Embedded & Hardware: Raspberry Pi 5, Arduino, PlatformIO, MCU firmware, motor drivers, LiPo power, UART
Tools: Embedded Linux, Git, GitHub Actions (CI/CD), unit testing, PyTorch, Inventor CAD, 3D printing