Jinay Patel

Robotics World Champion | jinayunity22@gmail.com | github.com/Github11200 | jinay-patel.vercel.app

Education

University of Waterloo
Bachelor of Computer Engineering

Experience

Perception Engineer September 2026 — Present
WATonomous *Waterloo, ON*

- Cut BEVFusion inference latency **25%** (43 to 32 ms) by adding ResNet-50 support, making camera processing 2.47x faster
- Built a two-stage PyTorch training pipeline that cached LiDAR features and fine-tuned 7.1M parameters on a single GPU
- Integrated a TensorRT/CUDA segmentation head that produces six-class road maps from camera and LiDAR inputs
- Removed 240,000 sigmoid evaluations per frame by thresholding segmentation logits directly in a CUDA kernel

Embedded Controls Software Engineer October 2021 — June 2026
Seaquam Robotics *Delta, BC*

- Achieved a **100% autonomous win rate** across 15+ matches with C++ PID control, odometry, and automatic corrections
- Increased movement accuracy by **200%** with errors under 1 inch by using a cubic-spline lookup table to tune PID constants
- Earned **3 Robotics Programming Awards** for innovative motion control algorithms and technical documentation
- Won the world championship Excellence Award and provincial championship Excellence award, recognizing overall robot design, programming, and competitive performance among **20,000+ teams** worldwide

Lead Software Engineer October 2023 — June 2026
Seaquam Robotics *Delta, BC*

- Led 4 developers in connecting awards, events, and history pages to a shared Deno backend for dynamic content updates
- Cut manual data entry by **50%** with a Deno and TypeScript backend aggregating team, event, and award data from 3 sources
- Reduced website updates from weeks to 60 minutes with a React and Tauri desktop content management system (CMS)
- Cut third-party API response time from roughly **500 ms to 100–200 ms** by caching responses in the Deno backend

Projects

Motion Control Framework | Embedded Controls Software Engineer GitHub

- Built drive-to-point PID and drive-to-pose lookahead controllers with odometry, drift compensation, and a 10 ms control loop
- Interpolated look up table that uses a Cubic spline to calculate PID gains prior to every movement
- Generated 2D cubic Bézier paths with 0.25-inch waypoints and acceleration limits; used RAMSETE with odometry based pose error to correct linear and angular velocity before feedforward wheel commands
- Built a custom self-registering C++ test harness with **89 tests** covering PID, geometry, Bézier curves, and interpolation

Verbosity: C++ Compiler | Personal Project GitHub, Blog Post

- Built a compiler with lexing, recursive-descent parsing, and C++ code generation for a custom language
- Implemented variables, functions, pointers, loops, and arithmetic with an AST and Pratt expression parser

Lynx (Automated Note Organizer) | Hackathon Project Obsidian Plugin Listing

- Launched a TypeScript Obsidian plugin using local Ollama models to organize notes, reaching **300+ users**
- Built retrieval-augmented search with LangChain vectors and a Graphology knowledge graph to answer note-based questions
- Synchronized vector and graph indexes as notes changed so answers reflect the most up-to-date version of the vault

Posturai | Hackathon Project - VSHacks 2nd Place GitHub

- Built a Next.js posture monitor with MediaPipe and browser-based ONNX inference for live webcam feedback and audio alerts
- Used shoulder-centered data augmentation to generate **2,000 pose samples** from 200 images, making the test accuracy go from **70%** to **81%**
- Trained a PyTorch classifier on **132 pose features**, then exported it to ONNX for browser inference
- Tracked sitting, break, and bad-posture intervals in local Dexie storage to power session analytics

Skills

Languages: C/C++ (5 years), JavaScript/TypeScript (6 years), Python (5 years), Rust (3 years)
Technologies: PyTorch, LangChain, Mastra, CMake, CUDA, React, Svelte, Next.JS, Tailwind CSS, Git, Docker, Linux, ROS2