

Abheek Pradhan

469-875-9205 | abheek.prad@gmail.com | linkedin.com/in/abheekpradhan | github.com/lolout1 | abheekp.dev

EDUCATION

Texas State University

Bachelor of Science in Computer Science

- Activities and Societies: Vice President - .EXE , ACM

San Marcos, TX

Expected May 2026

WORK EXPERIENCE

(UPCOMING PUBLICATIONS: MULTIMODAL HAR USING CROSS MODAL LEARNING ETA: 03/2026 ([LINK](#)))

Toshiba - Software Engineering Intern

May 2025 - September 2025

Toshiba International Corporation

- Developed STM32 multi-core FreeRTOS (real time operating system) firmware and TouchGFX UI for medium voltage drives on ARM/x64 architectures; improved C backend performance by **25%** through task priority optimization and stack size tuning using native FreeRTOS and CMSIS-RTOS2 HAL APIs (C , C++) following TDD principles
- Built automated testing infrastructure using **Python, Bash, and Ruby** scripting for 10,000+ parameter validation on hardware and simulators; created GitLab CI/CD pipelines for all tests (unit, integration, HIL (Hardware in Loop).
- Engineered RAG based **AI agent** for defect detection using Azure Copilot to parse CAD drawings from Oracle database, achieving **90%** accuracy in identifying manufacturing defects

Research Assistant (Texas State University)

August 2024-Present [Github](#)

- Developed multi-modal transformer and LSTM models for NSF-funded fall detection research under Dr. Anne Ngu; worked on multi modal and knowledge distillation pipeline in **Python** and deployed to edge devices in **Java** and **Kotlin**
- Deployed **PyTorch/TensorFlow** transformer models to edge (**Android**) via TFLite conversion using INT8 quantization and mixed-precision training, achieving sub-1% accuracy loss with 1.5x-2x battery life improvement on production devices
- Engineered **Python** multi-modal data pipeline with **PostgreSQL** database; implemented peak-finding algorithms and signal processing (Butterworth/Kalman filters, sensor fusion) for preprocessing, improving F1 score by **10%**
- Optimized inference on low-resource hardware (smartwatch, phone) via multi-threading, batch processing, operation refactoring, and hardware acceleration (GPU, NPU), delivering **3x** speedup when testing on mobile (phone , watch) in Java and Kotlin.

Machine Learning Engineer (Texas State University)

Dec 2024 - Sep 2025 [Try NeuroNest!](#) [Github](#)

- Collaborated with research team to fine-tune Vision Transformer and MASK R-CNN models on SLURM cluster with Nvidia A100 GPUs. Created a custom dataset achieving **98%** precision for Alzheimer's research object detection.
- Architected and deployed full stack infrastructure using **Python, Docker**, Hugging Face Spaces, and **AWS** services (S3, CloudWatch) for scalable backend Alzheimer's research platform serving real-time inference requests
- Optimized inference performance via multi-threading, RGB analysis, and parallel processing across **Python** backend, JavaScript , and **React** Native mobile application, reducing latency by **40%**

PROJECTS

FPGA Optimized Facial Recognition (C, C++, Embedded Linux, Yocto, PyTorch)

[YouTube](#) — [Github](#)

- Developed facial recognition on AMD Kria KV260 SoC achieving 99.47% accuracy with ensemble architecture for face detection, recognition, and landmark extraction using CUDA accelerated training and Vitis AI/Vivado toolchain optimization.
- Engineered zero-copy DMA architecture with hardware-accelerated GStreamer and INT8 quantization (16x model size reduction, sub 0.5% accuracy loss), reducing memory bandwidth by 60%.
- Delivered 100x CPU speedup and 300-800% industry improvement through multi-threaded processing and parallel DPU inference with custom Vitis kernels, improving throughput from 0.5-5 FPS to 30-500+ FPS.

Autonomous Chess Bot for online chess games (C++ , NodeJS , Javascript, HTML, Docker)

[YouTube](#) — [Github](#)

- Created a chess bot for online chess games integrating Stockfish engine API (3600 Elo) with Lichess.org using C++ TCP socket server, Windows IPC pipes, JSON messaging, and Javascript (Selenium , Powershell) automation.

Pricing Vanilla European Bitcoin Options (Java , Spring Boot , Python, Selenium, Javascript)

- Implemented Black-Scholes model in Java using Finite Difference Method and Monte Carlo simulation, with web scraping pipeline via REST APIs to validate theoretical Bitcoin option prices against historical contract data.

Real-Time Audio Transcriber (Python , Flask , ReactJS, Next.js, CSS, Git , GraphQL)

[Bitbucket](#)

- Created full stack agentic web app featuring a Flask WebSocket backend with React.js/Next.js frontend, achieving sub 500ms latency using hardware-accelerated multi-modal **LLM**.
- Designed asynchronous audio processing pipeline with MongoDB for session persistence and transcript storage.

SKILLS

Full-Stack: Python, C++, Java, AWS, JavaScript, TypeScript, UI/UX, NodeJS, Go, Ada, SQL, C#, Rest API, Kubernetes, Splunk, Git, Docker, NoSQL, MySQL, CI/CD, Selenium, HTTP, Qt, Jira, Agile, SCRUM, React

AI/ML: LLM, CNN, Computer Vision, Streamlit, MCP, MLOps, Spark, Transformers, Knowledge Distillation, .NET

Embedded Systems: FreeRTOS, VHDL, DSP, JTAG, Linux, I2C, SPI, UART, Drivers, Interrupts, VLSI