Abheek Pradhan

abheekp0@gmail.com | linkedin.com/in/abheekpradhan | github.com/lolout1 | abheekp.dev

Education

Texas State University *San Marcos, TX*

Bachelor of Science in Computer Science (Computer Engineering Concentration) *Expected May 2026*

- Activities and Societies: Vice President - ACM, IEEE

Publications: *(1st author) Dualstream kalman transformer Submitted: 01/26. Text 2 motion Diffusion - 06/26 (Link)*

Work Experience

Toshiba - Software Engineering Intern *May 2025 - August 2025*

Toshiba International Corporation

- Developed and mocked STM32 FreeRTOS firmware ; added and tested features for touchscreen interface on Toshiba MVDs
- Created DevOps testing infrastructure from scratch on x64 , ARM architectures (CMake , TDD , Python, Bash, Ruby), Built Jenkins CI/CD pipeline validating 10,000+ params via unit,integration, and HIL tests to cut QA time by 60%
- Optimized RTOS task priorities and DMA scheduling, reducing TouchGFX CPU overhead from 41% to 18% , achieving 25% total system CPU reduction , and eliminating all timing violations to correct issues revealed by my new HIL tests. (C, C++) .
- Engineered a RAG based AI agent using Microsoft Copilot , Azure , and OCR to detect defects in CAD pdfs with 94% precision

Research Assistant *August 2024-Present* ***Github***

Texas State University, NSF Funded

- Developed cross-modal distillation pipeline (video to IMU sensors) and personally trained custom multimodal transformers for time series forecasting under Dr. Anne Ngu. Deployed to edge devices (phones , wearables) with 92% F1 score in real time
- Automated data engineering and validation of 15,000+ sensor / video files via LLMs, DSP, and Computer Vision algorithms.
- Built distributed Ray Serve Slurm pipeline w/ automated testing of inference on edge devices achieving 1200% speedup, added efficient attention mechanisms increasing F1 +4%,tested DSP + sensor fusion algorithms to align modalities increasing F1 +5%
- Deployed multimodal PyTorch / TensorFlow transformer models to edge via LiteRT and ONNX using INT8 quantization + mixed-precision training, achieving 2-3x battery life w/ sub-1% accuracy loss allowing for on device inference (NPU , GPU)
- Refactored full-stack Android Studio app (Java / Kotlin) for ONNX, Kafka, Spark and MongoDB support.
- Built agentic dataset labeling framework for labeling 10,000+ samples of motion data to train text to motion diffusion models.

Machine Learning Engineer *Dec 2024 - Sep 2025* ***Huggingface*** ***Github***

Texas State University, Center of Analytics and Data Science

- Collaborated with research team funded by Texas State C.A.D.S to fine-tune Vision Transformer and MASK R-CNN models on distributed system Slurm cluster w/ Nvidia A100 GPUs . Created custom dataset achieving 98% precision for defect detection
- Built production backend REST API using Python FastAPI, PostgreSQL, and Docker for deployment on Huggingface; implemented server side async request handling and batch processing to handle concurrent React Native mobile app requests.
- Accelerated inference via layer fusion and ONNX to TensorRT engine conversion; reducing latency and cloud costs by 40%
- Built supervised learning computer vision labeling pipeline for images leveraging Detectron2, CVAT, and vision LLMs with MLOps, implemented active learning loop for low-confidence samples, reducing manual labeling hours by 80%.

Projects

Textbook2Video - 2nd Place Antler X Nvidia Hackathon (Python, MCP, React, Langchain) Huggingface - Live deployment

- Deployed agentic (LangChain) multimodal pipeline automating animated educational video gen from PDFs w/ ElevenLabs TTS, OCR, and(SFT) fine-tuning Llama LLM via LoRA + 4b quantization. Hosted with Docker container on HuggingFace

FPGA Optimized Facial Recognition (C, C++ 14 , Embedded Linux, Yocto, Vivado , PyTorch) YouTube — Github

- Finetuned computer vision models and deployed onto AMD SoC achieving 99.47% accuracy with ensemble architecture for face detection, recognition, and landmark extraction using Docker containerization for cross compilation on ARM64
- Engineered zero-copy DMA architecture with hardware-accelerated GStreamer pipeline, Vitis AI / Vivado toolchain optimizations, and INT8 quantization (16x size reduction, sub 0.5% accuracy loss), reducing memory bandwidth by 60%
- Delivered 100x CPU speedup and 300-800% via multi threading and paralell processing, improving throughput up to 10x .

Sortify - Full-Stack Document Management (Python , NextJS , RAG, PostgreSQL, React , LLM , Supabase , CSS) Bitbucket

- Built RAG pipeline using NLP and sentence transformers to generate vector embeddings for semantic PDF search with PostgreSQL pgvector database, hitting 94% retrieval accuracy through custom chunking algorithms using SOLID principles.

Distributed Chess Engine - Autonomous online chess bot (C++, HTML , Node.js , Docker, Selenium) YouTube — Github

- Built multi process C++ TCP / IP server with shared memory IPC and custom JSON protocol for real-time chess engine synchronization. Deployed via Docker + Electron javascript with browser automation pipeline; 100% live winrate online.

Skills

Full-Stack: AWS ECS, Go, Typescript, GCP, Javascript, Angular, Bash, OOP, Spark, SQL, C#, RESTful API, GraphQL, SEO, GitLab, Rust, Git, Docker, Spring Boot, NoSQL, Gradle, Selenium, HTTP, R, Qt, Jira, Agile, SCRUM, Unix
Deep Learning: LLM, CNN, Computer Vision, Streamlit, Matplotlib, MLOps, RLHF, scikit-learn, NumPy, Pandas, NLP
Systems & Embedded: FreeRTOS, UDP, DSP, JTAG, Linux, I2C, SPI, UART, Ethernet, Operating Systems, SDLC