Abheek Pradhan

469-875-9205 | 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

Upcoming Publications: *Dualstream kalman transformer , multimodal - ETA: 03/2026 (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 new touchscreen interface on Toshiba MVDs. Created automated 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 Azure Copilot and OCR for detecting defects in CAD drawnings with 94% precision.

Research Assistant (Texas State University , NSF Funded) *August 2024-Present* ***Github***
- 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 dataset cleaning and validation of 15,000+ sensor / video files via LLMs, DSP, and Computer Vision algorithms.
- Built distributed Ray Slurm pipeline with automated validation and testing on edge devices achieving 1200% speedup, added efficient attention mechanisms increasing F1 +4%,tested DSP + sensor fusion algorithms to align modalities increasing F1 +5%
- Refactored multimodal **PyTorch** / **TensorFlow** transformer models to edge via TFlite and ONNX using INT8 quantization + mixed-precision training, achieving 2-3x battery life w/ sub-1% accuracy loss allowing for on device inference (Python)
- Refactored full-stack **Android Studio** app (**Java** / **Kotlin**) to support **ONNX**, **AWS S3**, and **MongoDB** for analytics. Built secure distributed data pipeline with **Kafka** / **Spark** for async server-side inference and fault tolerant processing.

Machine Learning Engineer (Texas State University) *Dec 2024 - Sep 2025* ***Huggingface*** ***Github***
- Collaborated with research team funded by Texas State C.A.D.S to fine-tune Vision Transformer and MASK R-CNN models on distributed GPU Slurm cluster with Nvidia A100 GPUs . Created custom dataset achieving **98%** precision for defect detection
- Built production REST API using Python **FastAPI**, **PostgreSQL**, and **Docker** for deployment on **Huggingface**; implemented server side async request handling and batch processing to handle concurrent requests from React Native mobile app
- Accelerated inference via layer fusion and ONNX to TensorRT engine conversion; reducing latency and cloud costs by **40%**
- Built automated computer vision labeling pipeline leveraging Detectron2, CVAT, and vision LLMs for model-assisted labeling with MLOps pipeline, implemented active learning loop for low-confidence samples, reducing manual labeling hours by 80%.

Projects

Textbook2Video - 2nd Place Antler X Nvidia Hackathon (Python, GenAI, React, Langchain) Huggingface - Live deployment
- Deployed agentic (LangChain) multimodal pipeline automating animated educational video gen from PDFs w/ ElevenLabs TTS, Deepseek OCR, + 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
- Developed facial recognition on AMD Kria KV260 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 , Javascript , 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 , TCP , UDP , Java , GCP, Javascript , Angular , bash , OOP , Spark , SQL, C#, RESTful API, Kubernetes, GitLab , Gradle, Git, Docker, Express , MySQL, CI/CD, Selenium, HTTP, R , Qt, Jira, Agile, SCRUM, Unix
AI / ML: LLM, CNN, Computer Vision, Streamlit, MCP, MLOps, Spark, scikit-learn, Tableu, Pandas , NLP , Kafka
Embedded Systems: FreeRTOS, VHDL, DSP, JTAG, Linux, I2C, SPI, UART, Ethernet , Operating systems , VLSI