

AAKASH MADABHUSHI

+1 408-210-6122 | aakash.vardhan15@gmail.com | San Francisco Bay Area | Available Dec. 2026
linkedin.com/in/aakash-varadhan | github.com/aakashvardhan | aakashvardhan.github.io

SUMMARY

Train and evaluate models end to end, from data quality gates through PyTorch/CUDA training runs to measurement that holds up: 92% task success on a physical robot policy, 74.6% four-class accuracy from a network 30–50x smaller than its baselines, and a checkpoint-selection audit that caught a 67% worse policy before it shipped. Comfortable owning the pipeline that feeds a model as well as the model itself.

EXPERIENCE

Graduate Research Assistant

Mar. 2026 – Present

SJSU Applied Data Science Lab

San Jose, CA

- Established that validation-loss checkpoint selection would have shipped a **67% worse policy** (2.01 vs 1.20 action MAE) by training 52M-parameter Action Chunking Transformer and 99.9M-parameter SmolVLA policies in PyTorch/CUDA over 60K- and 20K-step runs and selecting on held-out action accuracy rather than on the validation-loss curve, which bottomed 13K steps before the better policy appeared.
- Reached **92% task success** and 2.49 cm mean placement error over 50 physical rollouts by deploying an ACT policy against an evaluation protocol locked before testing began, with each trial tagged no-grasp, grasped-then-dropped, or wrong-placement to separate perception failures from control failures.
- Benchmarked 18 ms against 251 ms inference latency across the two architectures to establish closed-loop feasibility, and traced 11–51 s policy stalls to out-of-distribution objects in the camera view.
- Cleared **49,633 training frames with zero integrity failures** by building pre-training quality gates over Parquet metadata and H.264 video that validate episode and frame alignment, decodability and resolution, and NaN-free action and proprioceptive tensors before a run starts.

Software Engineer

Jun. 2023 – Jun. 2024

Sapaad

Dubai, UAE

- Fed recommendation-model feature extraction with **5M+ rows per day** by shipping production Python ETL/ELT pipelines in Databricks and PySpark across a multi-tenant platform.
- Caught 14 schema, freshness, and ingestion anomalies automatically by building a Delta Lake data-quality framework in front of every pipeline feeding production ML, so training and serving data failed loudly rather than silently.
- Cut p95 dashboard query latency **43%** (12.4 s to 7.1 s) across 10+ Spark jobs serving 75+ tenant dashboards by profiling execution plans, rewriting SQL window functions, and placing targeted caches.

Computer Vision Engineer Intern

Sep. 2022 – Dec. 2022

Toothlens

Remote, India

- Turned architecture selection into a measured decision by defining a cycle-reconstruction SSIM protocol that compared candidates under partial occlusion, low light, and off-angle capture, then selecting CycleGAN over Pix2Pix and variational autoencoder baselines on the result.
- Removed the clinic's dependency on paired before/after patient captures it had no practical way to collect by building an unpaired image-to-image translation pipeline in PyTorch for orthodontic preview generation.
- Increased usable patient-image inputs **20%** across 1,500+ real-world photos by implementing OpenCV preprocessing guardrails that screened and corrected degraded captures before they reached the model.

PROJECTS

Multimodal Emotion Recognition | *PyTorch, CUDA, cross-attention fusion, Grad-CAM, SpecAugment* Feb. – May 2026

- Matched baselines **30–50x larger** at **74.6% validation accuracy** on four-class CREMA-D under an actor-independent split, by training a 1.57M-parameter audio-video network from scratch with cross-attention fusion between the two streams.
- Cut training epoch time **12x** (1,100 s to 88 s) by caching decoded media ahead of the training loop instead of re-decoding video and audio every epoch.
- Attributed every prediction to a specific facial region and frequency band using Grad-CAM over both modalities, turning a black-box classifier into one whose failures could be diagnosed.

U.S. Flight Delay Prediction at Scale | *PySpark, Spark MLlib, Kafka, Structured Streaming*

Dec. 2025

- Scored **0.711 F1** and 79.1% accuracy on a stratified 20% test split by training a Spark MLlib random forest over **9.5M flight records** drawn from 15 monthly BTS files totalling 3.1 GB.

- Held the train-minus-test accuracy gap to **0.3%**, against 1.6% for a single decision tree, by comparing ensemble against single-tree baselines on the same split rather than tuning to the test set.
- Moved the trained model onto a Kafka Structured Streaming path so each arriving flight receives a delay probability within seconds of ingestion.

Explainable Network Threat Detection | *PyTorch Geometric, Airflow, dbt, Snowflake, Llama 3.3* Feb. – May 2026

- Reached **0.90 F1** (0.94 precision, 0.87 recall) on 329 held-out host-pair links by training a graph neural network over a connection graph condensed from **27.1M Zeek network flows**.
- Delivered a plain-English rationale for every alert at **0.87 s** median end-to-end latency by grounding a Llama 3.3 explanation step in retrieved graph context.
- Built the Airflow, dbt, and Snowflake medallion pipeline that reduced raw connection logs into the host-pair features the model trains on.

TECHNICAL SKILLS

Machine Learning: PyTorch, CUDA, scikit-learn, Spark MLlib, PyTorch Geometric, transformers, graph neural networks, CNNs, cross-attention fusion, imitation learning, behavior cloning, GANs (CycleGAN, Pix2Pix), variational autoencoders

Training & Evaluation: checkpointed and resumable training runs, hyperparameter tuning, GridSearchCV, actor-independent and stratified splits, overfitting analysis, ablation studies, Grad-CAM interpretability, SSIM and F1 protocol design, Weights & Biases, Hugging Face Hub

Data for ML: pandas, NumPy, PySpark, Databricks, Delta Lake, Parquet, Airflow, dbt, Snowflake, data quality gates, feature extraction, SpecAugment and image augmentation

LLM & Retrieval: LangGraph, LangChain, Llama 3.3, Llama 3.2 Vision, Sentence Transformers, Pinecone, ChromaDB, retrieval-augmented generation

Languages & Infrastructure: Python, SQL, Bash, TypeScript, Docker, Git, Linux, FastAPI, GitHub Actions, AWS EC2 GPU instances

EDUCATION

San Jose State University
M.S. in Applied Data Science

Expected Dec. 2026
San Jose, CA

Illinois Institute of Technology
B.S. in Artificial Intelligence

May 2023
Chicago, IL