Chainathan Santhanam Sudhakar

Available: May 2025 – Dec 2025 | santhanamsudhakar.c@northeastern.edu | Portfolio | LinkedIn | GitHub | 857-746-2559

Education

Masters in Computer Science, Northeastern University, MA, **Grade: 4.0/4.0** Expected Graduation: Dec 2025

Bachelors in Computer Science, Vellore Institute of Technology, India, **Grade: 3.64/4.0** Aug 2018 – Jul 2022

Technical Skills

Languages: Python, MATLAB, OpenMP, MPI, JavaScript, Java

Frameworks: PyTorch, OpenCV, HuggingFace, YOLO, MediaPipe, Mathematica, Napari, ImageJ, Scikit-learn

Specializations: Deep Learning, **Computer Vision**, Gen AI, **LLMs**, **Vision Transformers**, Multi-modal models

Experience

Machine Learning Engineer Intern **Jun 2024 – Present**

Quantitative Neuroscience Lab - Northeastern University Boston, MA

- Developed a **Self-Attention U-Net** variant with multi-scale feature extractors for **Semantic segmentation of Neuron dendrites** with low SNR and high noise levels, enabling accurate dendrite structure masking.
- Designed a **Video stabilization** algorithm using PiV and trained a **U^{2}Net** model to **track protein** filament growth within dendrites via **kymographs**, delivering quantifiable insights into structural dynamics.
- Deployed **active learning strategies** to optimize labeling time by systematically selecting most informative samples.
- Engineered a high-performance **Napari app** for interactive dendrite manipulation and analysis, handling up to **50000 nodes**, incorporating advanced **Graph Theory** and **MATLAB**.
- Collaborated with a team of neuroscientists to streamline the dendrite analysis process, reducing processing time from **3 days to just 20 minutes** per dendrite – a **143x increase in productivity**.

Software Engineer **May 2022 – August 2023**

Societe Generale Bangalore, India

- Spearheaded the development of a Dashboard Web app enhancing KPI analysis, leveraging React, Spring, and microservices architecture, achieving a **2x productivity** increase for the middleware team.
- Executed data extraction and analysis from Datalake, applying complex business logic to streamline workflows.

Projects

Annotation of Object Positions in Rapid Motion Sequences [Code] | *Vision Transformers, MediaPipe* **Nov 2024**

- Engineered a pipeline to automatically **track** and **annotate** object positions in high-speed juggling sequences.
- Leveraged **MediaPipe** for robust hand tracking, **reducing annotation time by 85%** compared to manual methods to pinpoint time frames with initial and final object in hand reveals.
- Integrated **ViCLIP** with DETR (Vision Transformers) for precise object detection and hand classification.

Text to Music Generation [Code] | *Stable Diffusion, PEFT, Griffin-Lim* **Feb 2024 – Apr 2024**

- Developed a **real-time music generation** model by transforming **text prompts into Mel-spectrogram** images, converting **spectrograms to audio** with Griffin-Lim algorithm.
- Achieved a **3x speed up** over full fine-tuning on Stable Diffusion v1-5 by using **LoRA with PEFT on parallelized 2xT4 GPUs**.
- Reduced model checkpoint size from **7GB to 550MB** [adapter], saving **14x** storage in AWS S3 per checkpoint.
- Executed **latent-space interpolation** with image-to-image conditioning for enhanced **smooth audio transitions**.

Research Paper Summarization [Code] | *Gemma-7B, LLaMA-7B, Mistral-7B, QLoRA* **Feb 2024 – Apr 2024**

- Fine-tuned **Gemma-7B, LLaMA-7B, and Mistral-7B** on ArXiv ML papers, focusing on advanced data processing.
- Designed a hybrid **Extractive-Abstractive** pipeline to handle **long-document context** of more than 8k tokens.
- Leveraged **RoPE scaling** and **QLoRA** with half-precision enabling **2.4x batch** size while saving **50% GPU** memory.
- Deployed LLM with Continuous Batching, and Token Streaming by using Hugging Face open-source project TGI.

Image Generation [Code] | *Diffusion Models, Custom PyTorch Framework* **Dec 2023 – Jan 2024**

- Implemented **VAE, U-Net** and noise samplers like **DDPM, DDIM, PGDM** from scratch.
- Applied **Karras pre-conditioning** for faster convergence, achieving a **4.96 FID** on **LSUN bedrooms** dataset.
- Custom-built a Learner framework for flexible training pipeline for **gradient statistics**, metrics and experiments.