

Abhishek Mishra

mishraab@usc.edu | github.com/mishra077 | linkedin.com/in/abhishekmishra301202

EDUCATION

Master of Science – Computer Science
University of Southern California

Jan 2021-Dec 2022
3.57/4.0

B. Tech Computer and Communication Engineering
Manipal University, Jaipur

August 2016 - July 2020
9.19/10

SKILLS

Proficiency	Computer Vision, Deep Learning, Machine Learning, Image Processing, Natural Language Processing, GPU Profiling
Domain Knowledge	CNN 2D/3D, General Adversarial Networks, Transformers, supervised and semi-supervised Learning, PCA
Libraries & Frameworks	TensorFlow, Keras, Pytorch, OpenCV, NumPy, Sklearn, NLTK, Pandas, CUDA C++17, NVIDIA Nsight Compute, Pthreads, Open MP, Docker, Kubernetes, GitLab, GitHub, AWS, Git, Linux, EC2, S3
Programming Languages	Python, C++, C
Web Dev Tools	HTML, CSS, JavaScript, SwiftUI, NodeJS, Angular, PostgreSQL, Flask, REST, Jenkins, Databases, Apache Spark

WORK EXPERIENCE

Software Engineer | Cansera

Feb -23 - *

- Collaborated with healthcare professionals and engineers, harnessing **Rest APIs** and **Node.js** to gather and organize datasets efficiently.
- Effectively designed and implemented robust **data synchronization** processes utilizing automated cronjobs, ensuring seamless and scheduled data consistency, accuracy.
- Assisted in maintaining the healthcare dashboard and CLI (Command Line Interface) for Fitbit users. Utilized **Python scripting** to automate tasks and ensure a seamless user experience.

Research Intern | Aibee Inc

May 22 – Aug 22

- Collaborated with **Tracking and Re-Identification** team to improve the **feature extraction, embeddings** and overall performance of the pipeline using **Pillow, OpenCV, Pytorch, C++** and data parallelization.
- Engaged in innovative research and development with **Vision Transformer**, successfully adapting its capabilities to optimize cluster re-identification feature encodings.
- Achieved exceptional performance enhancements, surpassing the baseline model by a remarkable **2x** margin.
- Conducted a thorough analysis using **PCA, and T-SNE**, to understand hard outliers in the data samples.

Student Researcher | Interactive Lab, USC

May 21 – Aug 21

- Effectively implemented state-of-the-art **Object Trackers**, such as **SiamRPN, SiamRPN++, and ATOM**, meticulously assessing their performance metrics as stated in the corresponding research papers.
- Conceptualized and implemented a robust Spatio-Temporal Network seamlessly integrated into the RPN blocks of the SiamRPN++ Network, resulting in a substantial performance boost.

Research Intern | Vision Intelligence Lab, MNIT

May 18 – June 20

- Implemented multiple novel Deep Neural Networks using **Python, Keras and TensorFlow** that detects change (**Motion Segmentation**). Leveraged **Spatio-temporal** features to generate background models from consecutive frames, ensuring accurate identification of dynamic elements within the scene.
- Additionally, the networks excelled at real-time segmentation of the current frame into foreground and background, with foreground pixels effectively representing change or motion within the scene.
- Developed and implemented several advanced End-to-End Change Detection architectures, leveraging custom dataloaders for efficient data loading in multiple GPUs and **OpenCV** for preprocessing. Delivering exceptional real-time capabilities and achieving state-of-the-art performance.
- Designed these architectures with a focus on optimizing Inference Speed and Model Accuracy, resulting in low latency and high throughput during inference.
- Conducted testing and inferencing of various neural network architectures for ablation studies and comprehensive analysis.
- Proficiently explored diverse Deep networks, including **2D Convolution, 3D Convolution, RNN, LSTM, GAN**, and Capsule Network, showcasing versatility in handling various deep learning paradigms.

PUBLICATIONS

- **3DCD-UV**: A Scene Independent End-to-End Spatiotemporal Feature Learning Framework for Change Detection in Unseen Videos, IEEE Transactions on Image Processing. [\[Paper Link\]](#)
- **3DFR**: A Swift 3D Feature Reductionist Framework for Scene Independent Change Detection, IEEE Signal Processing Letters. [\[Paper Link\]](#)
- **Poster titled 3DCD**: A Modular Spatio-Temporal Feature Learning Framework for Change Detection, Nvidia GPU Technology Conference, Silicon Valley, USA, 2019. [\[Poster Link\]](#)

ACADEMIC and PERSONAL PROJECTS

Stock Market App | Finn hub API, SwiftUI, GCP, Node JS, Angular

- Developed a Native stock search IOS App which has search, buy, sell, favorites and portfolio functionalities. Backend is deployed on GCP. [\[Project Link\]](#)

Optimizing Monte Carlo Path Tracer | C++ 14, CUDA, Nvidia Nsight Compute

- Optimizing the execution time and GPU resource utilization of Peter Shirley's Path Tracing Rendering System for different scenes using CUDA programming and profiling tools to achieve a speed up of 1663x, with branch efficiency of 0.91 and an achieved occupancy of 0.58. [\[Project Link\]](#)

Character Based Neural Decipherment for Lost Languages | Python, Pytorch, Min-Cost Flow, Seq2Seq Model, NumPy, Pandas

- Researched on deciphering archaic languages using symbols of known language. Proposed model able to identify 81% cognates correctly between Ugaritic (lost) and Hebrew (known) languages, generated results within 15.13% improvement. [\[Project Link\]](#)

Checkers Game AI Playing Agent | C++ 14, Minimax, Alpha-Beta Pruning

- Created game playing for checkers using minimax algorithm and performed Alpha-beta pruning to increase the efficiency at greater depths by 70% with heuristics considering relative piece position, priority, piece importance, etc in optimal time. [\[Project Link\]](#)

2D CNN Semantic Segmentation | Python, TensorFlow, Keras, OpenCV, NumPy

- Trained 2D CNN model on CamVid/ Mapillary dataset and by using predefined pixel segmentation list of various class objects.
- Used pretrained weights of the network trained on ImageNet for classification in the encoder part of the network.

Emotion Detector | Python, OpenCV, Numpy, Pandas, Scikit-Learn

- Developed an Emotion Detecting Model using the Cohen-Kanade Dataset. The model used Cascaded Haar Classifier to localize the Face in the given Image, after which Eyes and Mouth were found on the face in a similar fashion.
- The Eyes and Mouth are then passed through a Histogram of Oriented Gradients (HOG) filter. The resulting output feature vectors are then passed on to a Support Vector Machine (SVM) to classify the image into these 6 emotions: neutral, happy, fear, anger, disgust, sadness.

Immersive AR Game with Interactive Visualization | AR Foundation, Unity, C#

- Developed a series of innovative AR projects utilizing AR Foundation, including interactive experiences such as Building Planets AR, Jet Turbine Simulation, Gardeni AR, and Furnish AR.
- Skillfully utilized AR Foundation's robust feature set to create intricate interactive simulations, showcasing expertise in real-time virtual object integration, physics-based interactions, and spatial UI design.