**Tolga Aktas**

(585) 733-1545 | tolga.f.aktas@gmail.com

**EDUCATION**

Rochester Institute of Technology August 2021 - Present

*Doctorate in Imaging Science*

* **Coursework:** Noise & Random Signal Modelling, Fourier Methods in Imaging, Radiometry, Human Visual System, Image Processing, Computer Vision, Deep Learning for Computer Vision
* **Research Area:** Lifelong Machine Learning, Self – Supervised Learning

University of Rochester August 2015 - *May 2020*

*Bachelor of Science Electrical & Computer Engineering, Minor in Computer Science*

* **GPA:** 3.6 out of 4.0
* **Relevant Coursework:** Deep Learning, Computer Vision, Reinforcement Learning, Digital Signal Processing, Image Processing, Autonomous Mobile Robotics, Stochastic Processes, Computer Audition, Data Mining, Detection & Estimation Theory
* **Tools & Skills:** C++, Python, Java, Tensorflow, PyTorch, ROS, CUDA C, OpenGL, C*,* Keras, MATLAB, Scikit-Learn, CMake
* **Online Learning:** *Deep Reinforcement Learning*, *AI For Healthcare,* Modern C++, Natural Language Processing, PyTorch DL Fellowship, Private & Secure AI, Neural Networks and Deep Learning, Convolutional Neural Networks, Sequence Models (Recurrent Neural Networks), Structuring ML Projects, Improving DNNs: Hyperparameter tuning, Regularization and Optimization

**ELECTRICAL ENGINEERING AND COMPUTER SCIENCE EXPERIENCE**

*Research Assistant, kLab, Rochester Institute of Technology October 2021 - Present*

* Investigating the use of SwinTransformer model as a general backbone for visual understanding and integrating it into REMIND model of Hayes et. al 2020 using **PyTorch**.
* Studying alternative self-supervised learning architectures for lifelong machine learning tasks.

*Software Engineering Intern, Google Geo 3D & Photogrammetry May 2020 – September 2020*

* Implemented a 3D reconstruction pipeline in **C++** for camera parameter estimation and dense point cloud generation towards building a high-fidelity textured 3D reconstruction from satellite imagery.
* Implemented image filtering algorithms in **Python** for texture processing, feature extraction and building contour detection, and super-resolution tasks on 30cm multi-spectral satellite imagery.

*Research Assistant, VISTA Lab at University of Rochester October 2019 – April 2020*

* Investigated different visual and text embedding models for improving the accuracy on Visual Question Answering (VQA) tasks.

Studied style-conditioned GAN model literature for high-fidelity image generation tasks.

*E5 Entrepreneurship Fellow, Ain Center of Entrepreneurship at University of Rochester September 2019 – April 2020*

* E5 is a competitive fellowship granted to U of Rochester students with an entrepreneurial mindset in order to pursue their entrepreneurial efforts as part of their undergraduate education.
* Lead a project that aims to design a wearables-based solution to provide real-time workout activity recognition and tracking.
* Implemented signal processing methods for cleaning ECG, EMG and accelerometer data and detecting salient biometric features for real-time exercising analytics.
* Designed different experiment settings for building an annotated dataset of ECG, EMG and accelerometer data to be collected during workout sessions.

*Software Engineering Intern, Qualcomm Technologies Inc.*

*Extended Reality (XR) Systems* May 2019 – September 2019

* Built OpenGL ES application in **C++** for avatar rendering using OBJ files developed in Autodesk Maya.
* Implemented eye-tracking algorithm in **C++** to integrate eye tracking capabilities
* Implemented facial landmark detection and tracking algorithm in **C++** to add real-time gesture tracking capabilities
* Worked on variational/conditional VAE to generate avatar facial texture images using **TensorFlow.**
* Investigated deep learning-based methods for alternative generation of occluded facial landmarks from speech and/or occluded image.

*Xerox Undergraduate Research Fellow, Robotics & AI Lab, University of Rochester* June 2018 – August 2018

* Explored hardware architecture models on different platforms to perform prediction and perception tasks more efficiently and faster for real-time robotic operations
* Implemented perception, mapping, localization and controller algorithms in **C++**, built a **ROS**-based software stack for mobile robotics.
* Set up experiments and built an image dataset of moving robots & objects collected from TurtleBot 2.
* Gained practical and theoretical experience in parallel programming and optimization on **CUDA C**.

**PUBLICATIONS**

* Songyang Zhang, **Tolga F. Aktas**, Jiebo Luo, “Mi YouTube es Su YouTube? Analyzing the Cultures using YouTube Thumbnails of Popular Videos”
* Ian Lawson, Gazi Naven, **Tolga F. Aktas** “Speech Accent Classification and Unsupervised Accent Clustering”

**PROJECTS**

*Fully Autonomous Mobile Robots:*

* Built a fully autonomous system on TurtleBot-II for CSC 232: Autonomous Mobile Robots final challenge. Extensively used **ROS**, **C++** and **Git** control throughout the semester-long project.
* Built modules on ROS runtime for simultaneous localization and mapping, perception, path-finding controller, *A\* search* path planner for shortest distance path to the goal points.

*DotLip: Lip Reading with Deep Neural Nets*

* Implemented a spatio-temporal CNN + recurrent neural network architecture in **Tensorflow** and **Keras** for performing lip-reading transcription from image and image + audio.

*Wavelet-based Statistical Speckle Reduction in Ultrasound Images*

* Implemented the statistical filtering methods in **Python** proposed by Gupta et. al (2004) for reducing speckles in medical ultrasound images.
* Implemented 12 filtering methods and evaluated their comparative performance on edge preservation and noise suppression.

*Speech Source Separation using Deep Neural Networks*

* Implemented fully connected and LSTM-based neural networks to separate human voice from the background audio waveforms in **PyTorch**.

*English Accent Classification from Speech*

* Implemented clustering algorithms to apply an unsupervised learning scheme to map different voice datapoints into different clusters.
* Implemented a Bidirectional LSTM with Attention mechanism to classify different English accents from speech in **PyTorch**

*Senior Capstone: Real-time Library Patron Monitoring*

* Implemented human body detection and tracking algorithms using **Tensorflow**.
* Built a full-stack solution using **Vue.js**, **NodeJS** and **MySQL** to process and store anonymized human body count and location for librarian personnel’s use.

***Online Learning Projects***

*Sequential Models and Applications*

* Trained a LSTM-based neural network by using both **Tensorflow** and my own implementation of LSTM for generating 30-second long Jazz-style audio files.
* Implemented RNN and LSTM networks from scratch for character-level language modelling using **Numpy**, for learning word embedding matrices, and debiasing the embedding model in terms of gender.
* Implemented and experimented with Attention networks for machine translation tasks, and a simple trigger word detection algorithm using **Tensorflow**.

*Convolutional Neural Networks with Applications*

* Implemented the forward & backprop algorithms for CNNs, auxiliary functions, residual convolutional networks using **Python**.
* Implemented the YOLO algorithm for a car detection and localization task, a CNN algorithm using **Tensorflow** for face recognition and detection tasks.

*Natural Language Processing*

* Studied part-of-speech (PoS) tagging, named entity recognition, stemming and lemmatization, stop word removal using **NLTK** toolkit. Built an elementary Bayes Spam Classifier to detect spam emails. Studied Hidden Markov Models (HMM), implemented Viterbi algorithm. Built a HMM for PoS tagging.
* Studied NLP feature extraction methods and paradigms such as bag of words, TF-IDF, word embeddings, word2vec, GloVe, DL-based embeddings, t-SNE. Studied and implemented topic modelling using Latent Dirichlet Analysis (LDA). Implemented and compared gradient-boosted DT classifier and RNN for sentiment analysis.
* Studied sequence-to-sequence encoder-decoder models, attention models, self-attention and transformers. Implemented word preprocessing, simple RNN, RNN with embeddings, Bidirectional RNN, and encoder-decoder RNN models for end-to-end machine translation from English to French using **Keras**.
* Created and deployed an Amazon Alexa skill using **Alexa Skills Kit (ASK)**
* Studied the fundamental topics in Automated Speech Recognition (ASR) such Fourier analysis, feature extraction with MFCC, DL for speech-to-text translation tasks, Connectionist Temporal Classification (CTC) loss function. Implemented and compared the performance of RNN, RNN + TimeDistributed Dense (TDD), CNN + RNN + TDD, Deep RNN + TDD, Bidirectional RNN + TDD neural networks using **Tensorflow** and **Keras** for end-to-end ASR task.

*Modern C++ Development*

* Built a system monitoring application like htop in practicing advanced OOP techniques.
* Studied modern memory management paradigms of C++ such as memory allocation types, dynamic memory allocation, memory copying policies, move semantics, smart pointers. Optimized the memory management of an example chatbot application using smart pointers and move semantics.
* Studied multithreading concepts such as running threads, data passing, mutexes and locks. Built a multithreaded traffic simulator using a real urban map

*Deep Reinforcement Learning*

* Introduction to Deep RL:
* Implemented Monte Carlo prediction for estimating action-value function, and constant – alpha MC control for estimating optimal policy.
* Implemented TD Control Sarsa, TD Control Q – learning (Sarsamax), Expected Sarsa algorithms for studying Time Difference methods.
* Implemented an RL algorithm in **Python** to train a taxi agent in OpenAI Gym’s Taxi-v2 environment to pick passengers and drop off at correct locations.
* Value – Based Methods:
* Studied Deep Q-Network (DQN) algorithm and relevant early literature, implemented DQN to solve **OpenAI Gym’s LunarLander** environment, using **PyTorch**. Further studied improvements on DQN such as Double DQN, Prioritized Experience Replay, Dueling DQN and implemented Double DQN algorithm.
* Implemented and trained a RL agent that uses DQN in **Unity ML-Agents** environment to navigate a flat world environment and collect reward points.