VIVEK KOODLI UDUPA

Apt.302, 220 Elm St, Clemson, SC - 29631
(864)643-9650 ⋄ vkoodli@g.clemson.edu ⋄ https://github.com/VivekUdupa ⋄ www.linkedin.com/in/vkoodli

EDUCATION

Clemson University *Expected Graduation - May 2019*
Master of Science in Computer Engineering Overall GPA: 3.54/4.0
Department of Electronics and Computer Engineering

Visvesvaraya Technological University *August 2013 - June 2016*
Electronics and Communication Engineering

TECHNICAL SKILLS

Programming Languages	Python, C++, C, Java, MATLAB
Libraries	NumPy, Pandas, Matplotlib, SDL 2.0
Software & Tools	PyTorch, Tensorflow, JMP, VisualStudio, Git, LaTeX

ACADEMIC PROJECTS

DeepLearning (PyTorch)

- Developed a Deep Convolutional Neural Network to detect and classify ten categories of bullying actions from given images. Data Augmentation and Batch Normalization strategies were used to overcome the problem of overfitting in the training phase. An accuracy of 94% was achieved.

Artificial Neural Networks (MATLAB)

- Developed a Multilayer Feed Forward ANN with learnable parameters for logistic activation function.
  - Learnable parameters over standard fixed parameters increased the efficiency by 30%
- Character Correction using Hopfield Network.
  - Partially distorted characters were matched to their nearest resembling English alphabets using Hopfield Network based on the principles of Hebbian learning.

Computer Vision (C)

- Convolution using mean filters and sliding windows for image smoothing
- Character recognition using Canny edge detection filters
- Improvised character recognition using thinning, end-point and branch-point detection
- Semi automated segmentation using active contours and Ranged image segmentation based on surface normals

Data Driven 2D Game Development (C++ and SDL 2.0)

- Designed a 2D game engine in C++
  - *Incorporated Object Pool, Factory, Observer and Singleton Design Patterns*
    Game features : Explosions, Collision Detection and developed Artificially Intelligent sprites
- Image Rendering in C++ GUI using C++ SDL 2.0 primitives

Advanced Data Structures (Python)

- Optimization of Bellman Ford Algorithm - Implemented SPFA algorithm in Python 3.0 which optimized the performance of Bellman Ford algorithm on random graphs by 70%.

RELEVANT COURSES

Computer Vision, Artificial Neural Networks, Analysis of Linear Systems, Non-Linear Controls, Statistical Methods I, Robot Manipulators, 2D Game design, Data Structures, Analysis of Tracking Systems, Deep Learning