Nrupatunga

https://nthere.dev

+91-9972716433
nrupatunga.tunga@gmail.com
github.com/nrupatunga

Research Interests: Deep Learning for Scene and Object Understanding

Education

- **Indian Institute of Science** — Bangalore, India
  Masters in Signal Processing; CGPA: 5.8/8.0 — *2010 - 2012*
- **Sri Jayachamarajendra College of Engineering** — Mysore, India
  Bachelor of Engineering in Electronics and Communication — *2005 - 2009*

Skills Summary

- **Languages**: Python, C, C++, Matlab
- **Deep Learning Frameworks**: Caffe, MXNet, PyTorch, Tensorflow
- **Tools**: Vim, Git, Tmux, Microsoft Visual Studio, QT, Eclipse, Android JNI/NDK

Experience

- **Whodat™** — Bangalore, India
  Deep Learning Research Engineer — *April 2017 - Current*
  - **Face Recognition for KYC automation**: Deep learning based system to verify the user's selfie with the photo in the KYC document and also to identify duplicates (if any) in the database to avoid fraud
    - Finetuned ResNet-100 single network & ResNet-100 sibling network with weight sharing. Both using additive angular marginal inter loss combined with intra-marginal loss
    - Implementation of other loss functions - Fixed & Dynamic AdaCos, Dynamic Weight Imprinting (DWI)
    - Achieved: ***TAR 94.32% @ FAR 1e-5***, ***TAR 89.95% @ FAR 1e-6*** with 4-fold cross validation
    - Languages & Tools used - Python, MXNet, Tensorboard, Visdom
  - **Egocentric Camera Height Estimation in Indoors**: Deep learning based system trained on synthetic dataset to estimate the height at which camera is held to calculate the real-world scale
    - Languages & Tools used - Python, Caffe, Visdom
  - **Ground/Wall plane and centroid estimation**: Deep learning based monocular depth, normal and segmentation to estimate the planes, their orientation and the centroid for placing the virtual objects in the real-world scenes
    - Theano to Tensorflow code conversion to achieve speed for both training and inference
    - Flask app to integrate with SLAM and also multiprocessing queuing system to handle multiple models.
- **Samsung R&D Institute** — Bangalore, India
  Technical Lead — *July 2012 - March 2017*
  - **Deep Convolutional Network for Food Recognition**:
    - Squeezenet model (accuracy=69%) ported on to mobile. ResNet, Inception models for better accuracy
    - Data collection, labelling and models to support Indian Food Categories
    - Languages & Tools used - Python, Caffe, Tensorflow
  - **Fully Convolutional Network for Segmentation of Sky and Non-sky regions**:
    - Fully convolutional VGG-16 model using SIFT flow dataset
    - Sky segmentation map used as prior for horizon detection in an image
    - Languages & Tools used - Python, Caffe
  - **Detection of Duplicate images in Gallery**: Nearest Neighbor Image Retrieval using GIST descriptor (Code)
    - Languages & Tools used - C++, OpenCV, Matlab
  - **One Touch Auto Image Enhancement (Commercialized in flagships after Galaxy S6)**:
    - Algorithm for detection of low-light/backlight, poorly lit face images
    - Architecture design of auto image enhancement engine
    - Languages & Tools used - C, Matlab
  - **Photo Editor/Best Photo (Commercialized in all android Phones)**:
    - Red eye correction, Blur Detection and Ranking Algorithm
    - Bilinear resizer module for less memory architecture in Photo Editor
    - Languages & Tools used - C, C++ Matlab
  - **Touch Focus (Commercialized in flagships after Galaxy S5)**: Complete JNI framework design & development for communicating between application and engine

Personal Projects

- **GOTURN single object tracking (Code)**: Python + Caffe based implementation to facilitate easy training and experimentation
- **Combining Sketch and Tone for Pencil Drawing Production (Code)**: Color pencil sketch effect for images which mimicks human style of pencil drawing. Languages & Tools used - C++, OpenCV, QT
- **Pedestrain detection using Histogram of Oriented Gradients (Code)**: SVM model to detect pedestrains in the image. Languages & Tools used - C++, Python, OpenCV
- **Image Processing Toolbox (App)**: QT based GUI application to test basic blur and edge detection algorithms. Languages & Tools used - C++, QT, OpenCV

Honors and Awards

- Awarded Employee of the month - Jan 2016
- Awarded Galaxy S5 for the effort in success of Touch Focus USP

References

- Upon Request