Nrupatunga

https://nthere.dev

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

Research Interests: Scene and Object Understanding, Metric Learning

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, PyTorch-Lightning, TensorFlow
- **Tools**: Vim, Git, Tmux, Microsoft Visual Studio, QT, Eclipse, Android JNI/NDK

Experience

- **BYJU'S (Think & Learn Pvt. Ltd.)** — Bangalore, India
  Staff Research Engineer — *September 2019 - Present*
  - **Handwriting OCR for Worksheets**: End to End pipeline design and development
    - Developed preprocessing algorithm to normalize the input to the OCR model
    - Developed a NN classification model to distinguish kids' interactions: writing, occlusion, small/bigger character
    - Frame by frame logic to optimize the number of OCR runs per frame
    - Languages & Tools used - Python, PyTorch
  - **Page Number Tab detection and orientation estimation for Worksheets**:
    - Developed keypoint based object detection/estimation network
    - Improvement of 12.5% in worksheet page number detection and 16-25% in timing compared to previous methods.
    - Languages & Tools used - Python, PyTorch, Visdom
  - **Page Boundary estimation, Page number and Page Id OCR for Worksheets**:
    - Formulated page boundary estimation as keypoint detection and association of corners using Part Affinity Field maps to form the boundary
    - Modification of data pipeline for OCR and network architecture to improve the robustness of the recognition
    - OCR precision improved by 2% with similar performance and less memory foot print on device
    - Languages & Tools used - Python, PyTorch, Visdom
- **Whodat™ (merged with BYJU'S)** — Bangalore, India
  Deep Learning Research Engineer — *April 2017 - August 2019*
  - **Face Recognition for KYC automation**:
    - Finetuned ResNet based single & sibling network with weight sharing using additive angular marginal inter loss combined with intra-marginal loss
    - Other loss functions- Fixed & Dynamic AdaCos, Dynamic Weight Imprinting (DWI)
    - Achieved: TAR 94.84% @ FAR 1e-5, TAR 90.69% @ FAR 1e-6 with 4-fold cross validation
    - Languages & Tools used - Python, MXNet, Tensorboard, 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
    - Languages & Tools used - Python, Caffe, TensorFlow, Theano, Flask
- **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-1)/(Code-2)**: PyTorch / Caffe implemenation to facilitate easy training and experimentation
- **Fast Image Filters with CNN (Code)**: Implementation of few Image filters using CNN
- **Combining Sketch and Tone for Pencil Drawing Production (Code)**: Color pencil sketch effect for images which mimicks human style of pencil drawing
- **Pedestrain detection using Histogram of Oriented Gradients (Code)**: SVM model to detect pedestrains in the image
- **Image Processing Toolbox (App)**: QT based GUI application to test basic blur and edge detection algorithms

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