Brain Corp 2019 Summer Intern - R&D Machine Learning Project

By Anirban Sinha, Stony Brook University

The submission includes two python scripts and training dataset folder.

- The file “train_phone_finder.py” is used for training a simple SVM classifier with nonlinear kernel which takes training data path as an input argument. During the process of training this program automatically generates a file named “anno.txt” that includes top-left and bottom-right corners of image patches to be used as positive examples. At the end of training, the trained parameters are written in a file named “finalized_model.sav”.
- The file “find_phone.py” is used to detect phone in a given input image. In the due process this program automatically reads learned parameters from the file named “finalized_model.sav”. If a phone is detected by “find_phone.py”, its normalized coordinated is returned and printed in the terminal.

The programs use following packages to accomplish the assigned task.

- Numpy, Cv2, Sklearn, Matplotlib, Pathlib, Pickle

The project is tested as functional with

- Anaconda3 (python3)
- Ubuntu 16.04

Future steps to improve "Phone Detector"

- Currently a simple SVM binary classifier has been implemented to detect phone from an input image. In order to make the classifier robust, hard-Negative-Mining approach to has been employed to augment the training dataset.
- Currently while generating positive or negative examples or even at the detection stage, no scaling of the images or image patches have been considered. In future I would like to extract features from magnified or compressed images for training so that classifier can recognize small and big size of phone in images
- In future I would also like to add feature to the phone detector that can detect multiple phones in one image or detecting phone from its partial view

Note on data collection

- Instead of center of the phone in a normalized coordinate, it would be better to provide information of the image patch that includes a phone described by top left and bottom right corner pixel coordinates