Project: Hand gesture recognition using Deep Learning

The purpose of this project is to create a system that allows no-contact interaction between a human and computer. This enables users to conveniently control machines

First step is to figure out a model that is trainable. Trainable means it achieves an accuracy of at least 80%, but >90% is desired. This is achieved by replicating other work done by people on already existing data.

Next step is to find/create an RGB dataset to train the model

* Only RGB pictures does not provide enough information to harness the characteristics of the hand, so an attempt to identify and isolate the hand in RGB images is done using image processing
  + NOTE: OpenCV reads image channels in a different order (BGR)
* Converting the RGB image to the HSV color plane allows fairly decent segmentation of the palm with some noisy background pixels, represented as a binary image
  + Segmentation lower and upper bounds are noticably different depending on background lighting. Therefore, a bright environment is assumed
* Erosion then dilation is applied on the binary image to reduce noise and emphasize the palm
  + Gaussian filter yields similar results to the dilation operation for binary images
  + Intensity thresholding simply does not work on binary images

Next, we test performance of our model on the new dataset

**Limitations**: An RGB camera will be used to acquire data. Therefore, training data must be in RGB of the same dimensions as the camera’s captured images (480x640)

(Block Diagrams for a high level structure is good)

**Progress:**

10/19, Trained on 160 images. 20 Epochs

* On a 15% test split, training achieves 97.8% accuracy but tests predict 67% images accurately
* Because the model can achieve high accuracy in training only, this must mean the test set has images that isn’t covered by the training set due to the random split OR the model got overtrained.
* It took about 10 epochs to reach a training accuracy > 90%.
* Therefore, we need a much larger for better performance.
* TODO: Gather more data
* TODO: Write code to display model’s confusion matrix
* TODO: Train model w 10 epochs and evaluate result