# Computer Vision - Hand Gesture Recognition

Shourya Aggarwal - 2017CS10379
Saksham Dhull - 2017CS10370

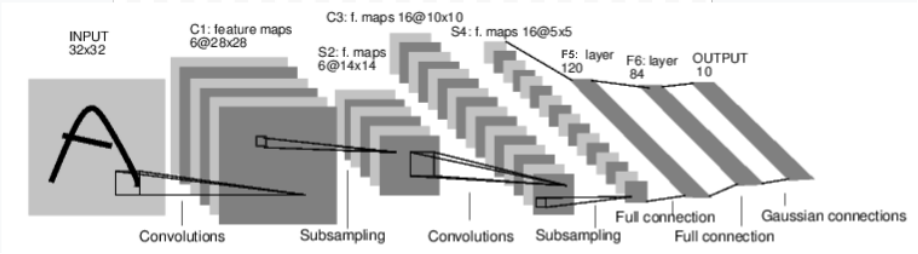
November 13, 2019

Training set and results

# 1 Introduction

## 1.1 Model Description

The convolution neural network model used for the assigment is:

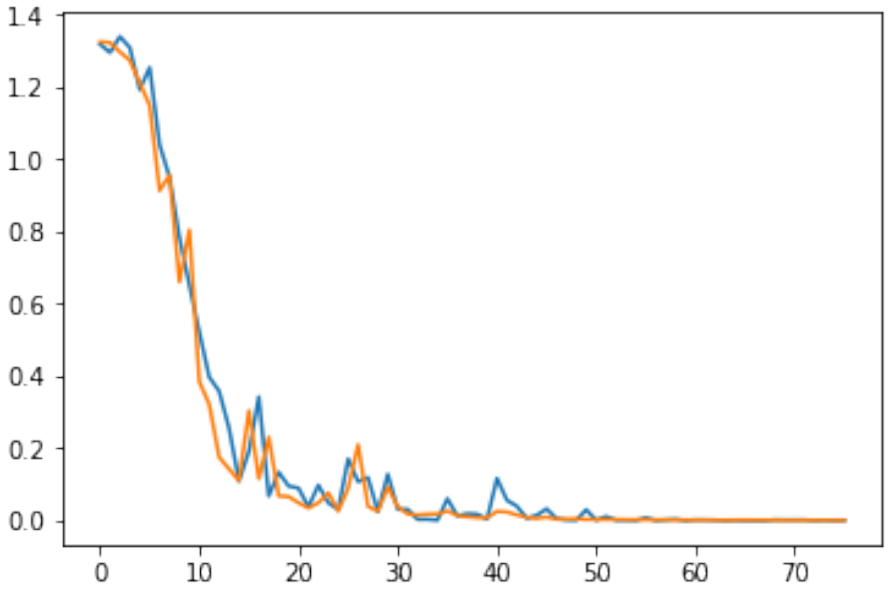



- `Conv2d(1, 6, kernel-size=(3, 3), stride=(1, 1))`: This is a convolution layer with $3 \times 3$ kernel size and 1 input image channel and 6 output image channels. There is no particular reason for using this format other than that it was similar to the one used for mnist.
- `Conv2d(6, 16, kernel-size=(7, 7), stride=(1, 1))`: This is a convolution layer which takes 6 channels input and outputs with 1296 features to the Fully connected layer.
- `Fully connected layers:` It consists of 3 layers, first with 1296 inputs and 500 outputs second with 500 inputs and 100 outputs and 3rd with 100 inputs and 4 outputs (final result classes). The neurons in each layer were taken so as to linearly decrease the size of feature map to 4.

Learning rate used was 0.001. This was a suitable rate as increasing this rate caused some fluctuations in learning graph. The data set was shuffled initially to prevent overfitting to one particular model and all the epochs were run on the same data from then on.

## 1.2 White background

- **Pre-processing:** For the training of hand gestures on white background, we used the images augmented with the corresponding contours drawn on them, as the training dataset of images. The contours were found out using simple Canny edge detection.
- **Inferencing:** The live frames taken from webcam were similarly augmented with the contours found out by Canny edge detection, and these modified frames are given to the trained model for inferencing.
- **Loss Graph:** Following is the plotted training loss (blue) and validation loss (orange), for the batches trained on.




## 1.3 Generic Background

- **Pre-processing:** For the training of hand gestures on generic background, we used colour segmentation to isolate the hand gesture in the frames, from the rest of the background. A pre-decided range of HSV colours were speficied to generate a binary mask of image pixels falling in that range, accompanies by gaussian blurring and morphological opening to smooth the image and reduce the white pixel noise from the mask. These masks are obtained from all the images from training data-set and the model is trained on these masks.
- **Inferencing:** The live frames taken from webcam were similarly masked between the range of skin colour to obtain the relevant masks. These

masks obtained are then given to the trained model for inferring. Sometimes the lighting conditions result in the perceived colour of hand skin to go out of the range during live infer mode, for which we have given the option to re-calibrate the colour range for current camera conditions.

- **Loss Graph:** Following is the plotted training loss (blue) and validation loss (orange), for the batches trained on. Note that some over-fitting is observed.