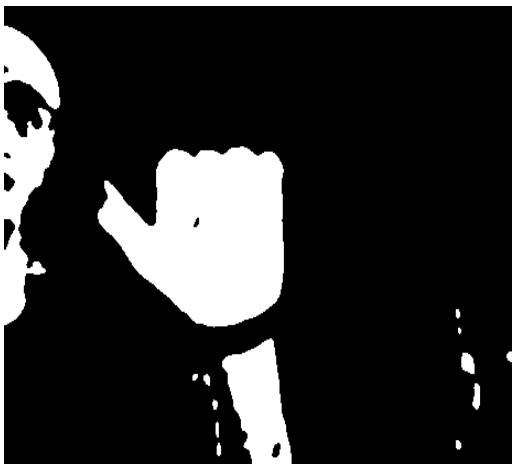
Assignment – 4
camera based music player control

Ayush Verma -2016CS10374 & Kartik Kumar - 2016CS10294

1. Preprocessing of Data -

A) Edge detection – We stated with training our model with edge detection but we later dropped this idea because for generic background it was unable to detect edges only of hands.

B) Background Subtractor and Hand detection – We started with detecting hand by making a mask of hue, saturatuion and lightning to extract human skin color and hence extracting the hand. As below



Next

Prev

Stop

2) Data saving-

Above method is implemented preprocessed images are converted to 50*50 and shuffled so that the images while batching to avoid overfitting to the batches. And then we have pickle dumped this in a file to read later while training.

3) Pytorch Model -

Layers -

A) Convolution layer

- **(a)** **Conv 2d –** input_channel =1 kernel_size = (5,5) output_channe=8 stride, padding = 0
- **(b)** **Maxpool layer -** kernel_size =(5,5) stride =2
- **(c)** **Relu activation function.**
- **(d)** **Maxpool layer -** kernel_size =(5,5) stride =3
- **(e)** **Relu activation function.**
- **(f)** **Conv 2d –** input_channel =8 kernel_size = (5,5) output_channe=16 stride, padding = 0

B) Dense layer

- **(a)** **Linear layer –** input size = 400 & Relu activation function
  output size = 128
- **(b)** **Linear layer –** input size = 128 & Relu activation function
  output size = 32
- **(c)** **Linear layer –** input size = 32
  output size = 4

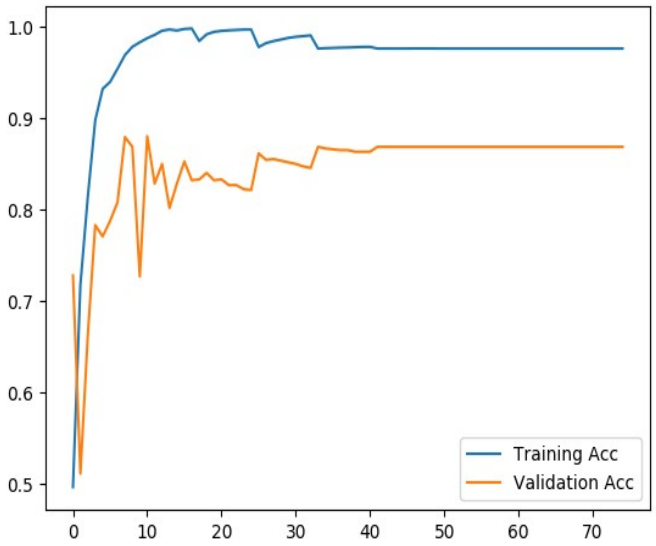
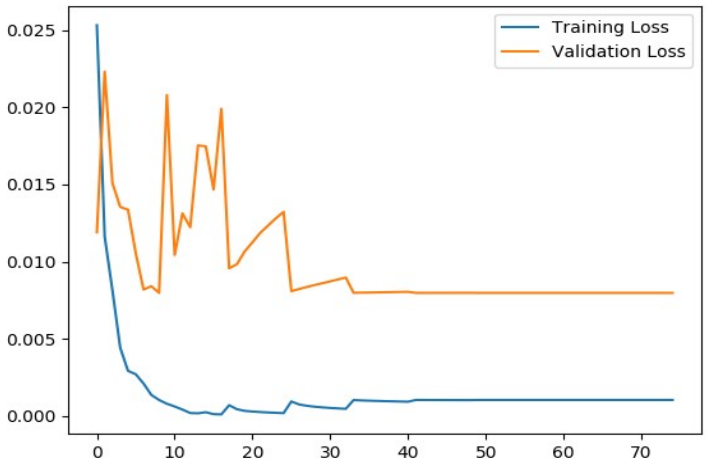
4) **Hyperparameters -**

- **(a)** Learning rate – Initially lr = 1e-3 but decreases if val accuracy does not decreases.
- **(b)** output channel – for Convolution layer we have tried to generate more output channel to capture different features.
- **(c)** Kernel size – As output channels are large so we decided kernel size of (5,5) to decrease number of trainable parameter.
- **(d)** Stride-To reducetainable parameter stides are set to 2or3
- **(e)** Loss function- Cross entropy loss.

(f) Optimizer- Adam optimizer.

(g) Early stopping is a form of regularization used to avoid overfitting on the training dataset.

5) Graphs -

Training loss and Validation loss with epoches. T Acc = 97.65 V Acc = 86.89	
Training loss and Validation loss with epoches. TL = 0.001048 VL = 0.007979	

References -

[1] https://pytorch.org/docs/stable/index.html
[2] https://arxiv.org/pdf/1901.10323v3.pdf
[3] https://github.com/ahmetgunduz/Real-time-GesRec
[4] https://github.com/Bjarten/early-stopping-pytorch
[5] https://github.com/khshim/pytorch_mnist
[6] Stackoverflow
[7] Wikipedia
[8] https://discuss.pytorch.org/
[9] Discussion of Assignment with Ansh Prakash and Adithya Anand