DATA603

Project 2

Shrey Patel | UID: 117583915

Shrey Nair | UID: 113160714

Introduction

This project serves as a research and evaluation for different architectures in Neural Networks and choice of parameters in them to enhance the performance of classifiers at every task.
Two types of data sets were used to perform tasks such as identifying subjects from pose.mat dataset and identifying handwritten digits in the MNIST dataset, respectively. After dividing the datasets into training and testing, we calculated the accuracy.

Task 1: Identifying subjects

I. Pre-processing

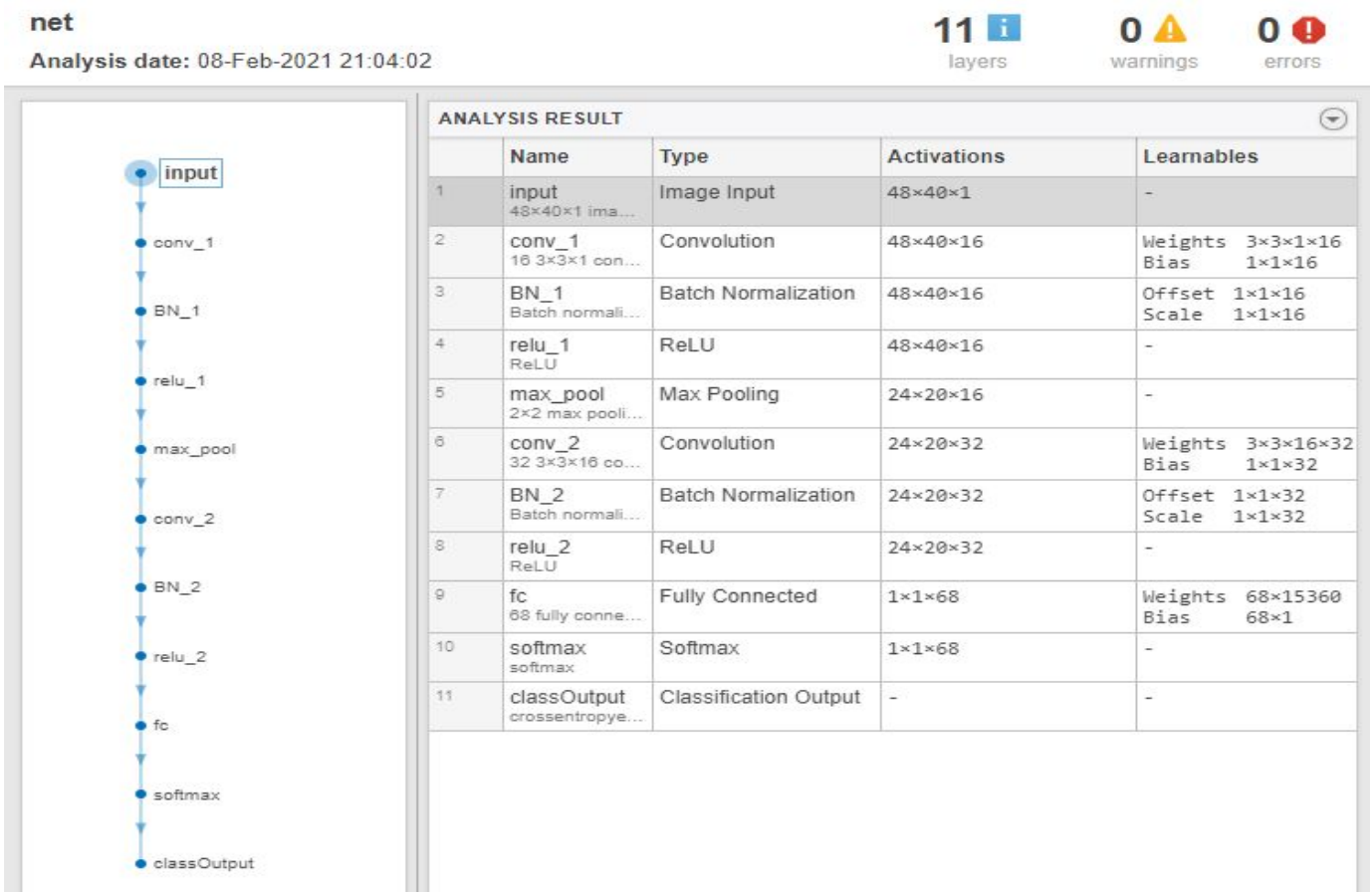
Here, pose.mat was used as the data set. The dataset consists of cropped images of 68 subjects. The train/ test split was performed in such a way that a training set was formed with first 10 poses for each subject, and a test set out of the remaining 3 poses of each subject.

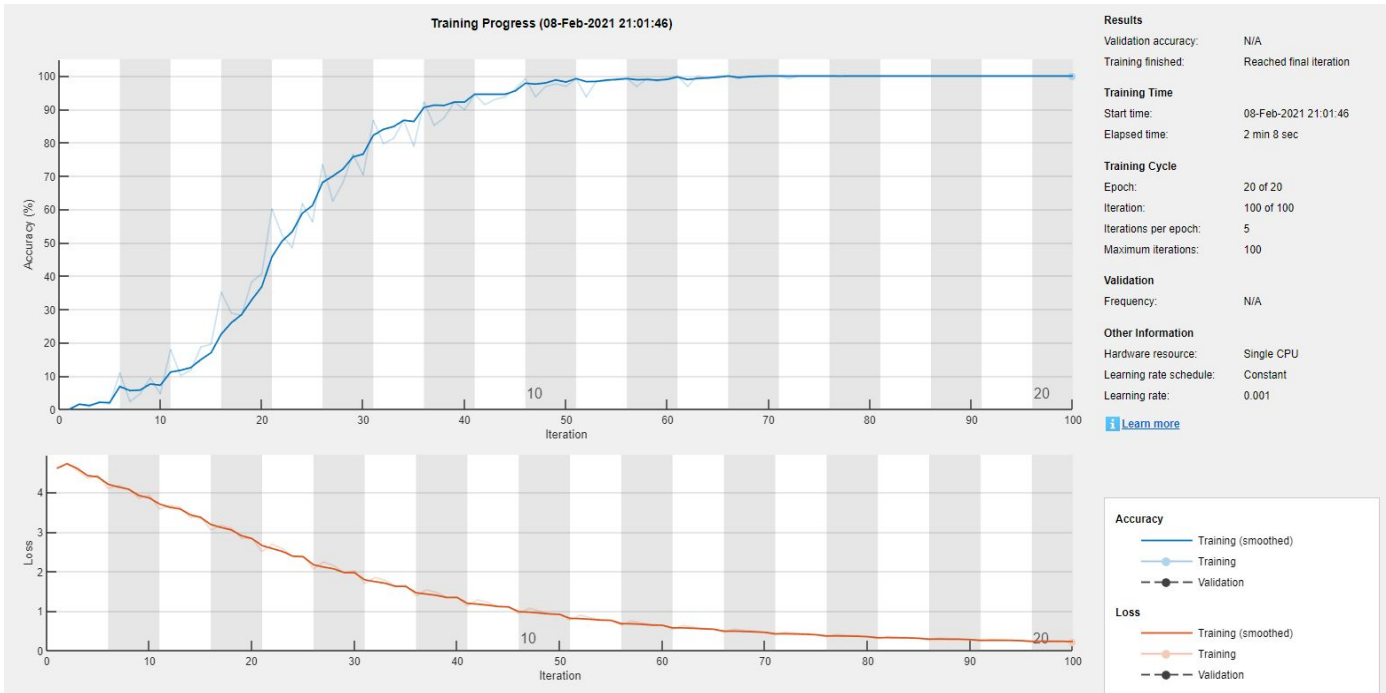
II. Train Neural Network

DeepLearning Toolbox was used to train various architectures of neural networks for deep learning using the trainNetwork function in MATLAB.

a. Convoluted Neural Networks

To start-off, a simple convoluted neural network was used to tackle this image classification problem. Further information regarding the layers can be found in the below figure.
The initial accuracy rate using the below convoluted neural network was **60.29%**.

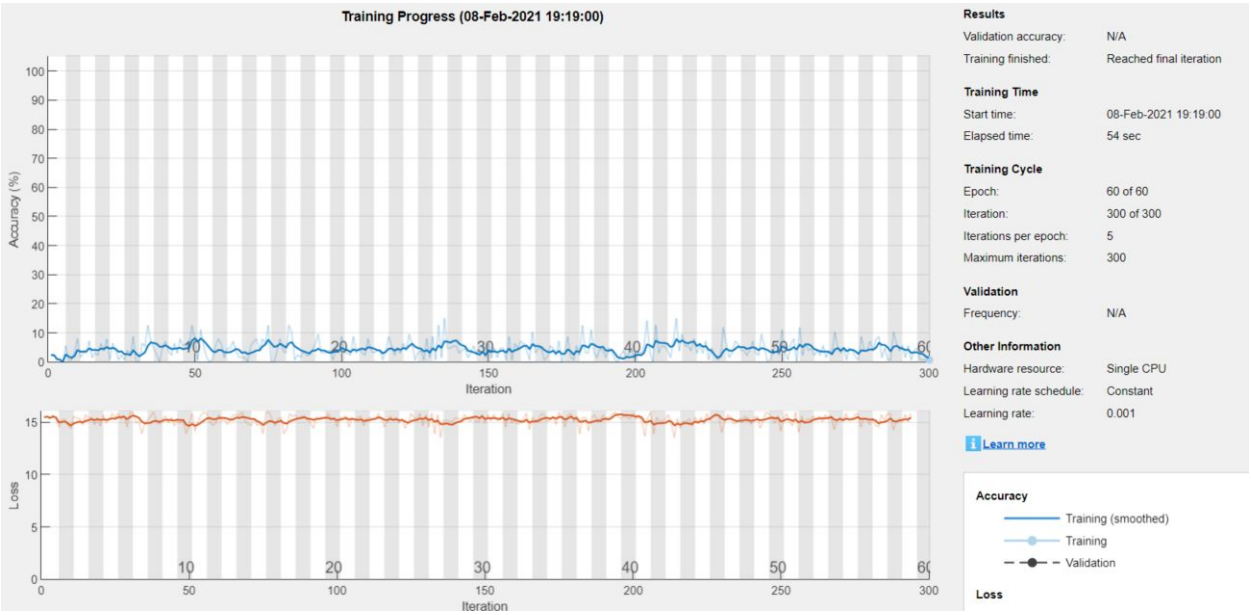




In order to improve the accuracy, further tests were performed:

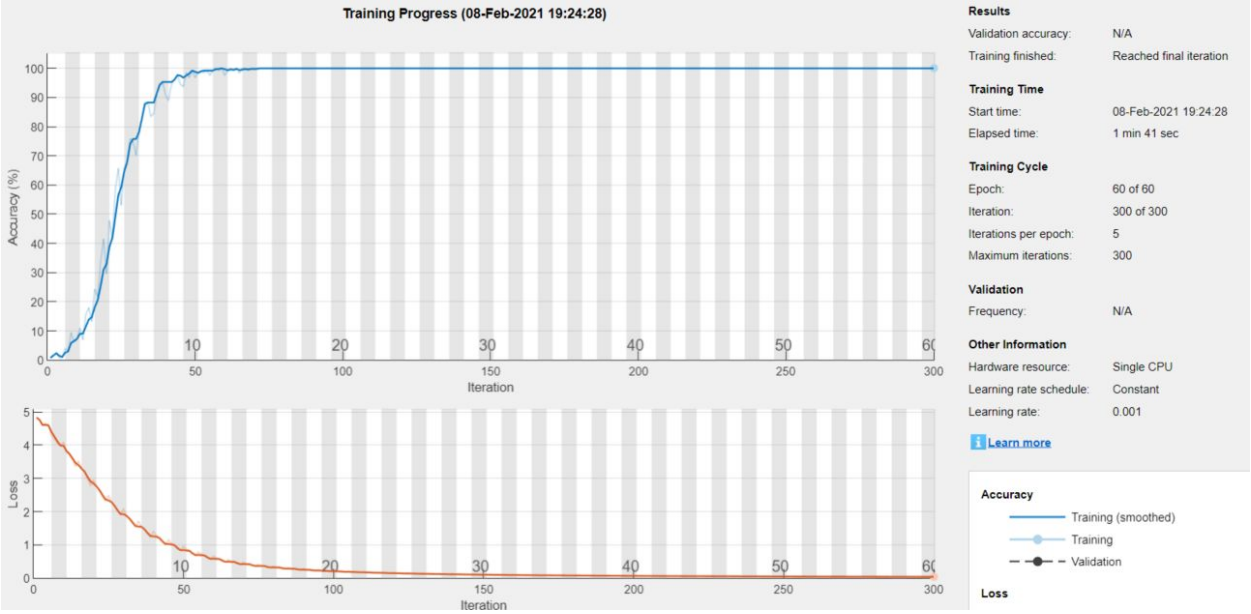
Test Case 1

- Parameters:
    - convolutionLayer with a filter size of 5 and 20 filters
    - ReLu layer to perform a threshold operation to each element of the input where any value less than zero is set to zero
    - Epochs were set to 60
    - Learning rate set to 1e-3
- Observation
    - The accuracy didn't seem to improve for each Epoch and the loss function was very high where a decreasing function is expected.



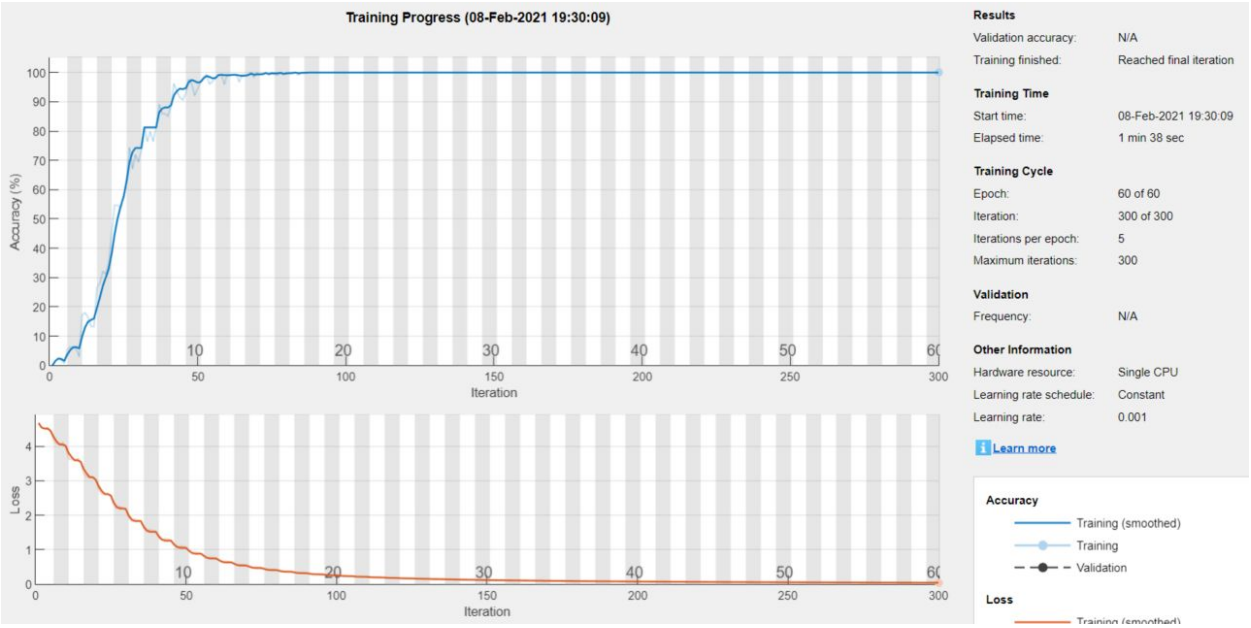
Test Case 2

- Parameters
    - convolutionLayer with a filter size of 3 and 16 filters with a padding of 1
    - 1st layer was batchNormalizationLayer and stride
    - 2nd layer was Relu Layer
    - Same Epochs and learning rate as attempt 1
- Observation
    - The accuracy seems to be improving over an increasing number of Epochs and loss function is decreasing.
    - Accuracy **63.24%**



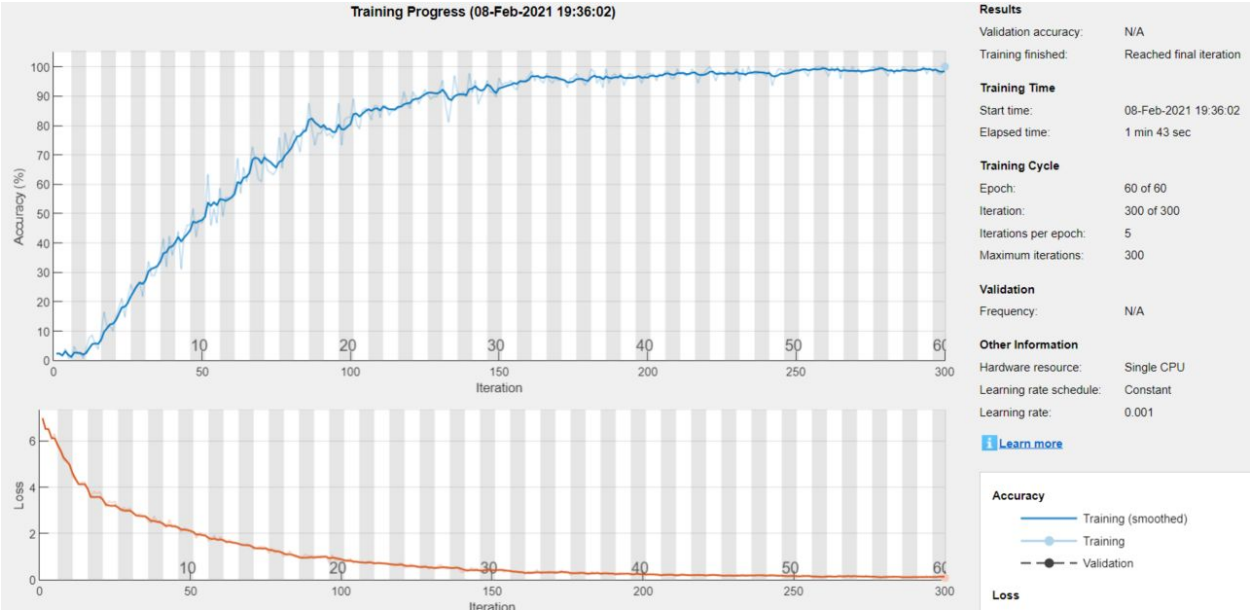
Test Case 3

- Parameters
  - First convolution layer was set with '5' filter size and 12 number of filters along with batchNormalizationLayer.
  - Tweaked the convolution layer for 2nd layer in the neural network with filter size of 3 and 32 filters along with relu layer
- Observation
  - Accuracy : **66.18%**.



Test Case 4

- Parameters
  - 1st layer is a convolution layer with filter size of 5 and 16 filters
  - 1st layer has relu layer, batchNormalization and stride
  - 2nd layer is a grouperConvolutionLayer with filter size 3, 24 filters and padding of 2 followed by relu layer, batchNormalization layer and stride of 2.
- Observation
  - Accuracy: **58.33%**.



b. Simple Directed Acyclic Graph (DAG) Neural Network

A DAG Network is a neutral network architecture for deep learning where layers are in the form of a directed acyclic graph. Below figure shows the further information regarding the layers in this architecture.

The accuracy obtained from this architecture is more than what we obtained from simple convoluted neural networks test cases. The accuracy score using this architecture for deep learning on the poses dataset resulted in **65.19%**, which is almost ~10% increase in accuracy in less number of iterations/epochs.

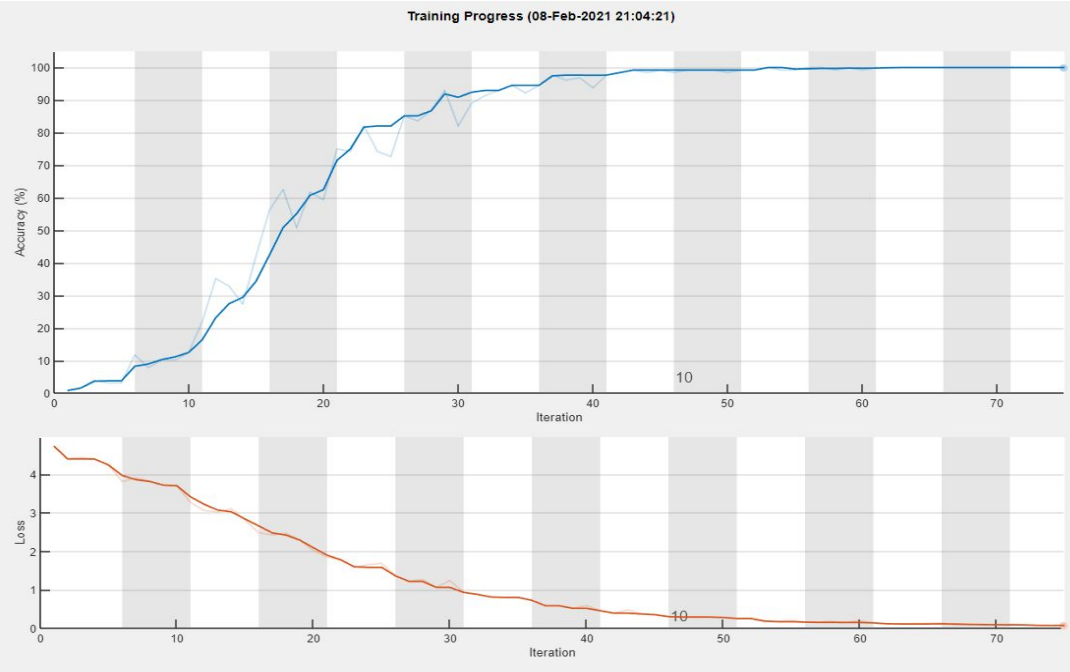
net

Analysis date: 08-Feb-2021 21:07:55

14 layers | 0 warnings | 0 errors

ANALYSIS RESULT

	Name	Type	Activations	Learnables
1	input 48×40×1 ima...	Image Input	48×40×1	-
2	conv_1 16 5×5×1 con...	Convolution	48×40×16	Weights 5×5×1×16 Bias 1×1×16
3	BN_1 Batch normali...	Batch Normalization	48×40×16	Offset 1×1×16 Scale 1×1×16
4	relu_1 ReLU	ReLU	48×40×16	-
5	conv_2 32 3×3×16 co...	Convolution	24×20×32	Weights 3×3×16×32 Bias 1×1×32
6	BN_2 Batch normali...	Batch Normalization	24×20×32	Offset 1×1×32 Scale 1×1×32
7	relu_2 ReLU	ReLU	24×20×32	-
8	conv_3 32 3×3×32 co...	Convolution	24×20×32	Weights 3×3×32×32 Bias 1×1×32
9	BN_3 Batch normali...	Batch Normalization	24×20×32	Offset 1×1×32 Scale 1×1×32
10	relu_3 ReLU	ReLU	24×20×32	-
11	avpool 2×2 average ...	Average Pooling	12×10×32	-
12	fc 68 fully conne...	Fully Connected	1×1×68	Weights 68×3840 Bias 68×1
13	softmax softmax	Softmax	1×1×68	-
14	classOutput crossentropye...	Classification Output	-	-



Task 2: Identifying handwritten digits

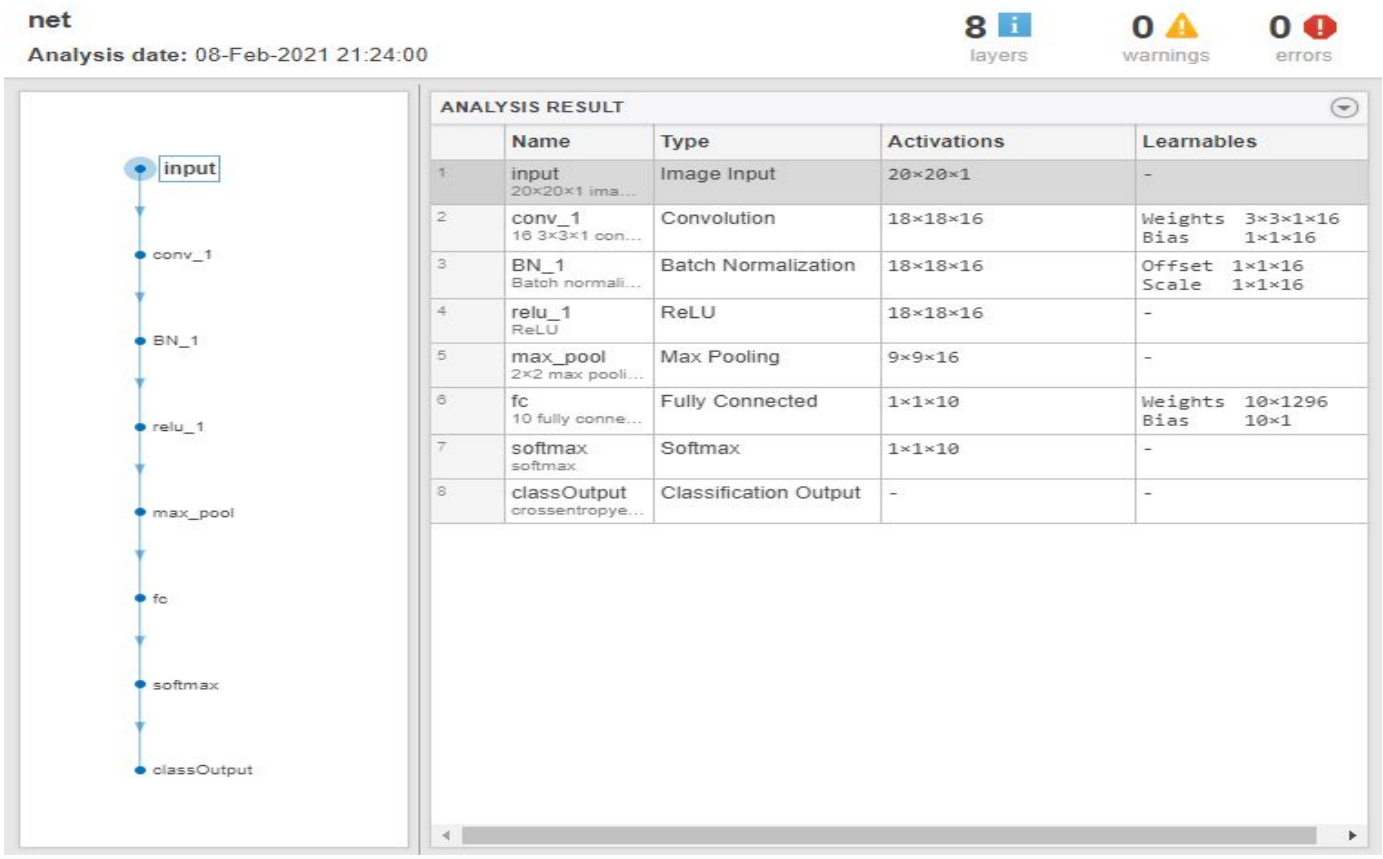
I. Pre-processing

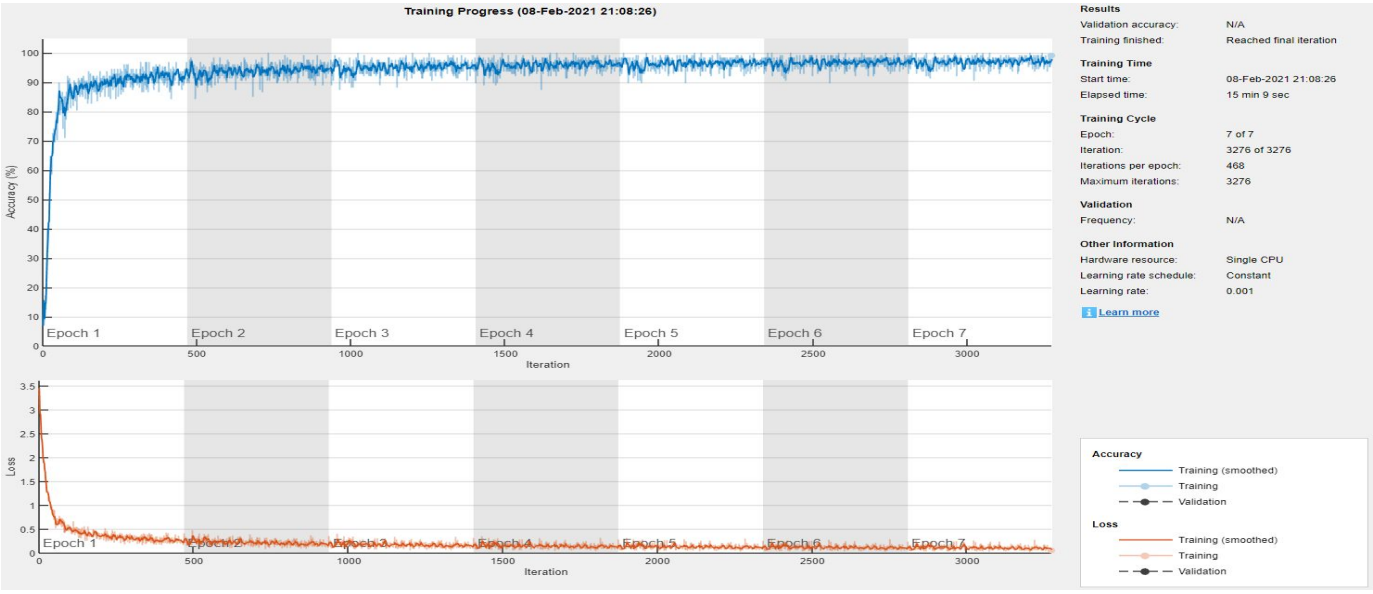
Here, mnist.mat was used as the data set. A script developed by Siddharth Hegde from Matlab file exchange was used to split the entire dataset into training set 60000 (28 x 28) grayscale images of handwritten digits (10 classes) and a testing set with 10000 images along with training and testing label set.

II. Train Neural Network

a. Convoluted Neural Networks

Similar to the last section, a simple convoluted neural network was used to tackle this image classification problem. Further information regarding the layers can be found in the below figure.
The initial accuracy rate using the below convoluted neural network was **97.03%**

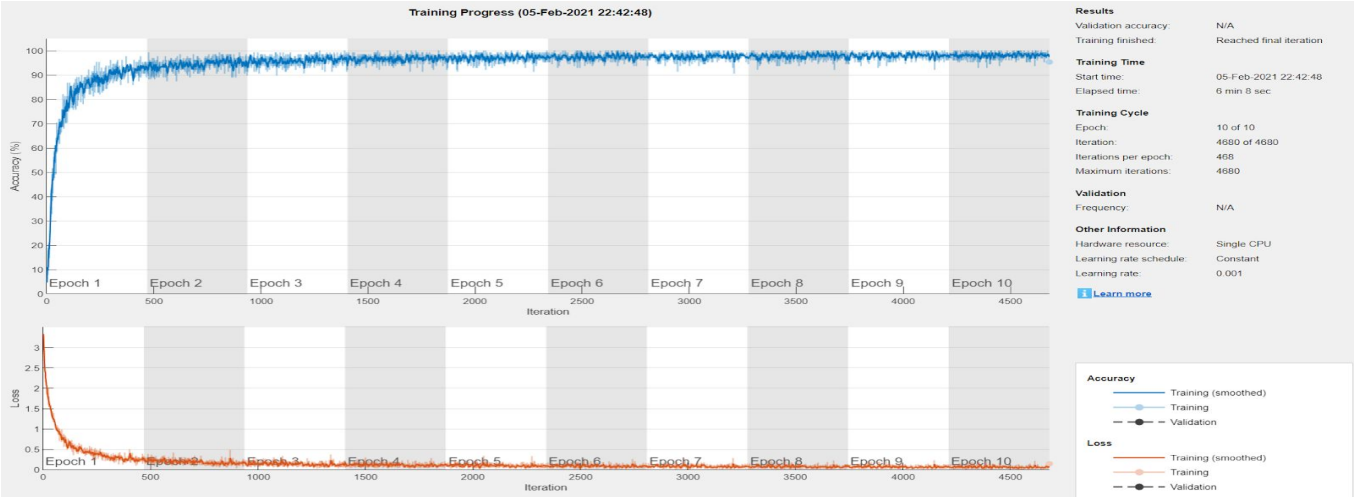




In order to improve the accuracy, further tests were performed:

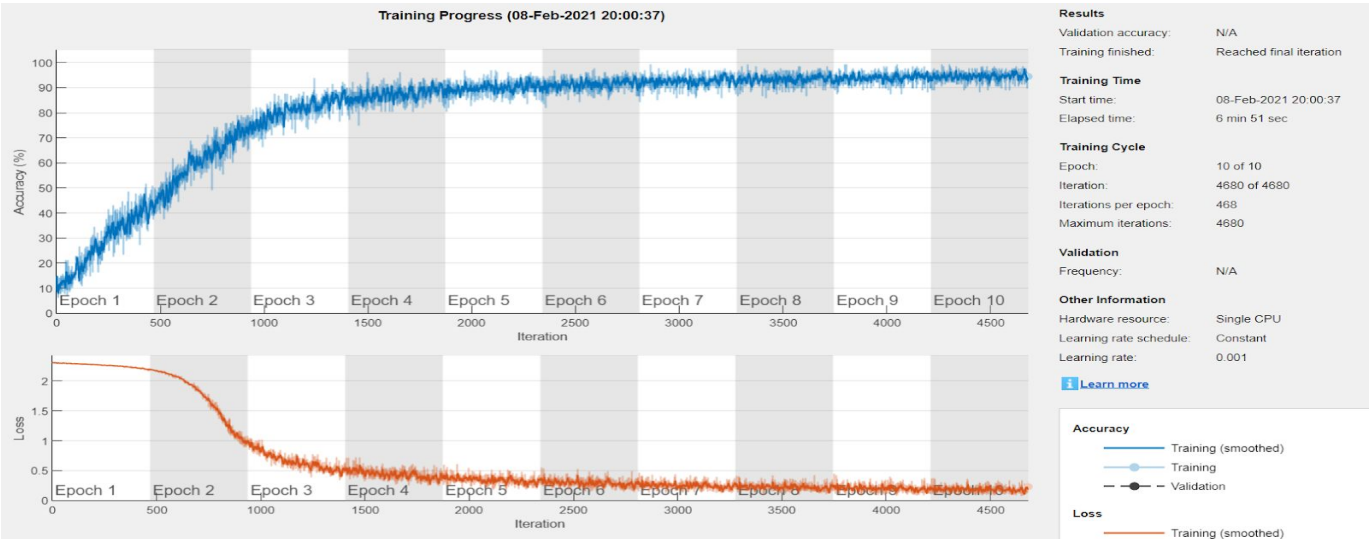
<u>Test Case 1</u>

- Parameters
  - 1st layer is a convolution layer with filter size of 5 and 16 filters and 1 padding followed by Relu layer and batch normalization layer
  - 2nd layer is a grouper convolution layer with filter size of 3, 24 filters and 2 padding followed by Relu layer and another relu layer
  - Epchs : 10
  - Learning rate: 1e-3
- Observation
  - Accuracy: **<u>98.64%</u>**



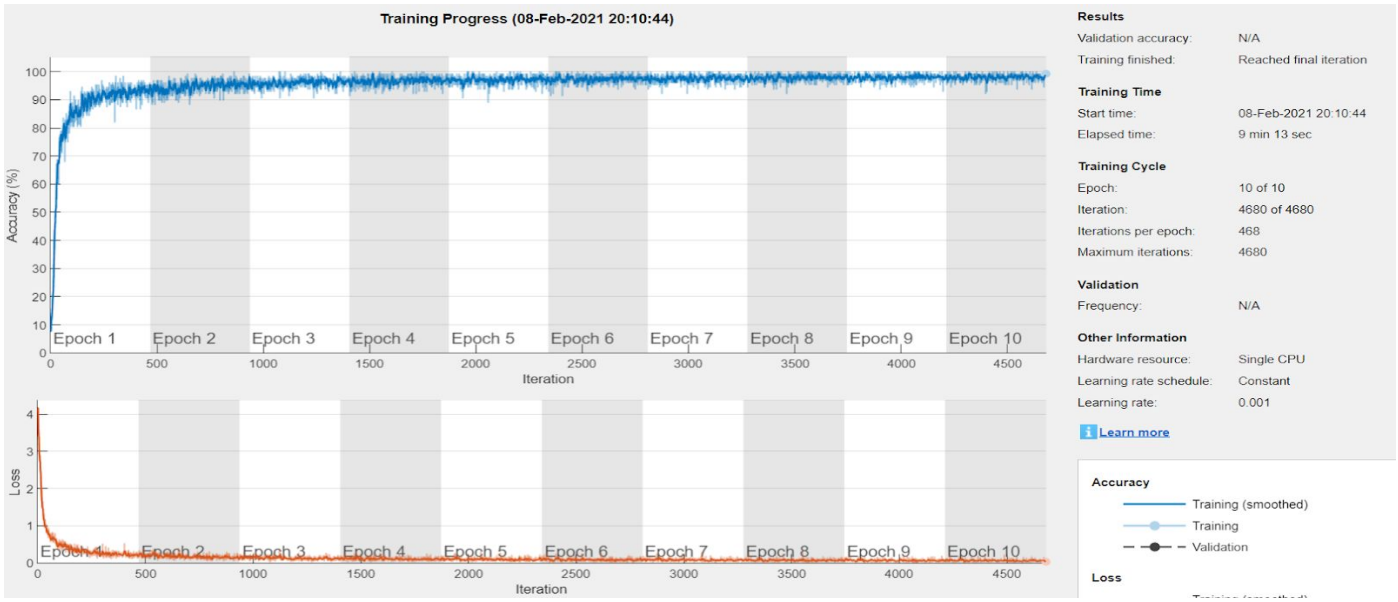
Test Case 2

- Parameters
  - 1st layer is a convolution layer with filter size of 5 and 16 filters and 1 padding followed by Relu layer and cross channel Normalization layer
  - 2nd layer is a grouper convolution layer with filter size of 3, 24 filters and 2 padding followed by Relu layer and cross channel Normalization layer
  - Epchs : 10
  - Learning rate: 1e-3
- Observation
  - Accuracy: **<u>95.98%</u>**



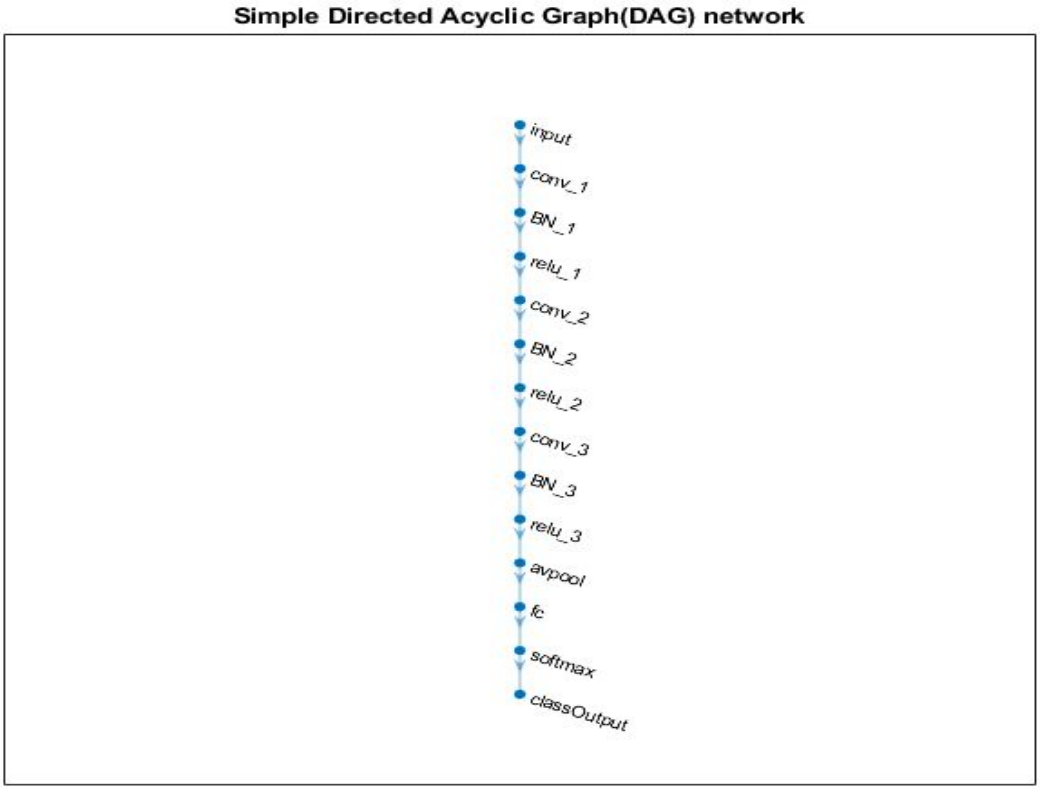
Test Case 3

- Parameters
  - The 1st layer is a convolution layer with a filter size of 3 and 16 filters and 1 padding followed by a Relu layer and batchNormalizationLayer.
  - The 2nd layer is a grouper convolution layer with filter size of 3, 24 filters and 2 padding followed by batchNormalizationLayer and cross channel Normalization layer.
  - Epchs : 10
  - Learning rate: 1e-3
- Observation
  - Accuracy: **<u>98.61%</u>**



b. Simple Directed Acyclic Graph (DAG) Neural Network

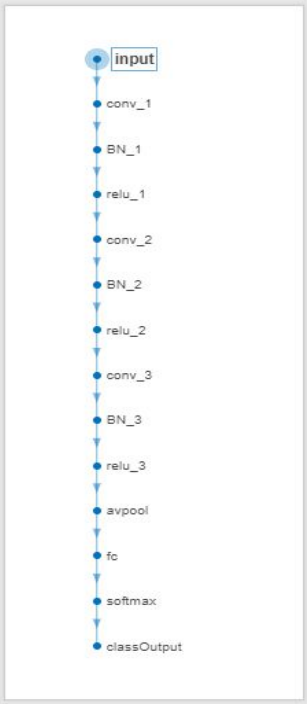
The accuracy obtained from this architecture is more than what we obtained from simple convoluted neural networks test cases. The accuracy score using this architecture for deep learning on the poses dataset resulted in **99%** in the same number of epochs.



net

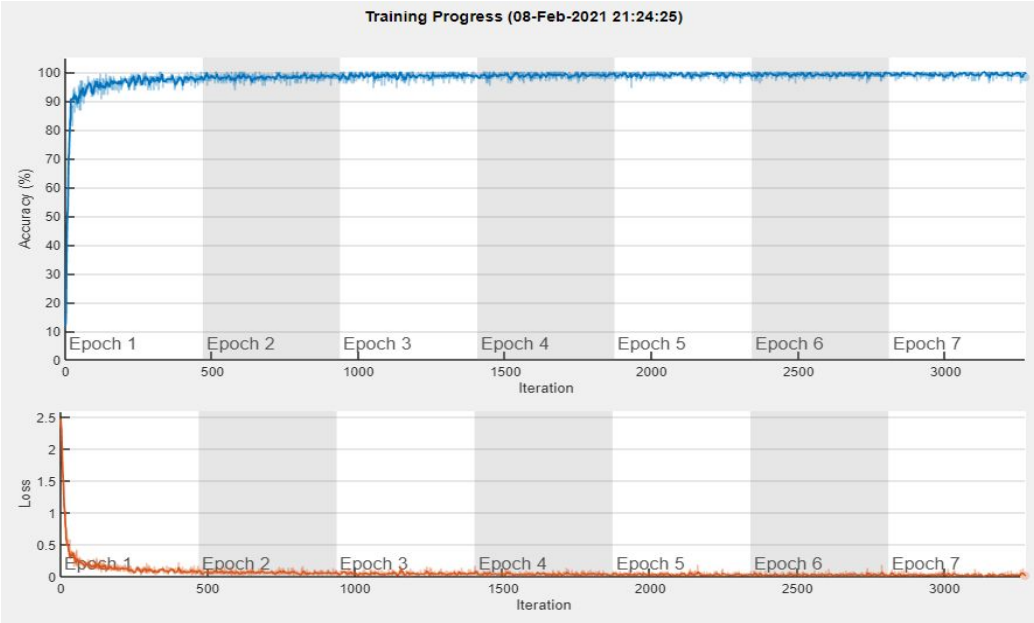
Analysis date: 08-Feb-2021 21:48:56

14 layers | 0 warnings | 0 errors



ANALYSIS RESULT

	Name	Type	Activations	Learnables
1	input 20×20×1 ima…	Image Input	20×20×1	-
2	conv_1 16 5×5×1 con…	Convolution	20×20×16	Weights 5×5×1×16 Bias 1×1×16
3	BN_1 Batch normali…	Batch Normalization	20×20×16	Offset 1×1×16 Scale 1×1×16
4	relu_1 ReLU	ReLU	20×20×16	-
5	conv_2 32 3×3×16 co…	Convolution	10×10×32	Weights 3×3×16×32 Bias 1×1×32
6	BN_2 Batch normali…	Batch Normalization	10×10×32	Offset 1×1×32 Scale 1×1×32
7	relu_2 ReLU	ReLU	10×10×32	-
8	conv_3 32 3×3×32 co…	Convolution	10×10×32	Weights 3×3×32×32 Bias 1×1×32
9	BN_3 Batch normali…	Batch Normalization	10×10×32	Offset 1×1×32 Scale 1×1×32
10	relu_3 ReLU	ReLU	10×10×32	-
11	avpool 2×2 average …	Average Pooling	5×5×32	-
12	fc 10 fully conne…	Fully Connected	1×1×10	Weights 10×800 Bias 10×1
13	softmax softmax	Softmax	1×1×10	-
14	classOutput crossentropye…	Classification Output	-	-



Results

Validation accuracy:	N/A
Training finished:	Reached final iteration

Training Time

Start time:	08-Feb-2021 21:24:25
Elapsed time:	23 min 49 sec

Training Cycle

Epoch:	7 of 7
Iteration:	3276 of 3276
Iterations per epoch:	468
Maximum iterations:	3276

Validation

Frequency:	N/A

Other Information

Hardware resource:	Single CPU
Learning rate schedule:	Constant
Learning rate:	0.01

Learn more

Accuracy

Training (smoothed)
Training
Validation

MATLAB Code

Shrey Patel : Link to code

Shrey Nair : Link to code