AMERICAN INTERNATIONAL UNIVERSITY BANGLADESH(AIUB)

COMPUTER VISION AND PATTERN RECOGNITION

Name	ID	SECTION
Shopnil, Md. Samiul Maleck	18-36540-1	A

MID TERM PROJECT REPORT

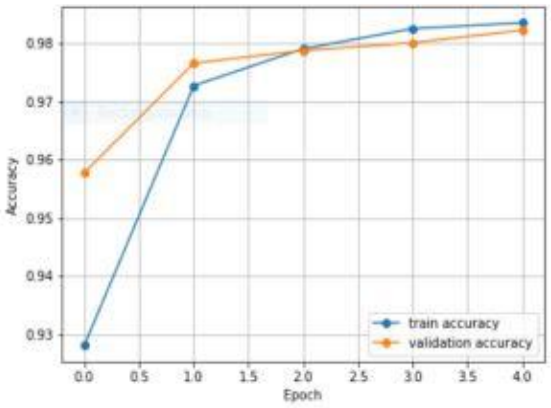
Abstract:

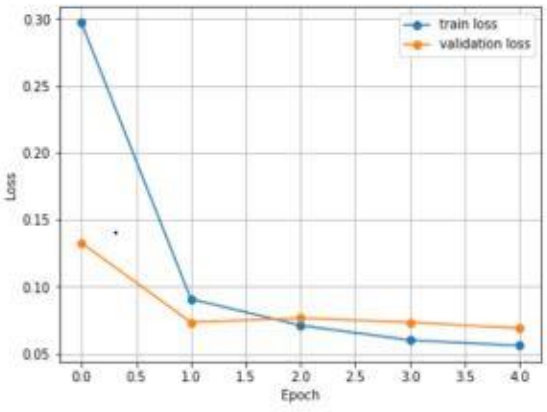
The Convoluted Neural Network (CNN, or ConvNet) is a class of artificial neural networks, commonly used for visual analysis. These are a kernel or filter based on the shared-weight architecture of Convolution that slides along input features and provides translation equivalents known as feature maps. In this report I have written a detailed dataset about the implementation of CNN architecture to classify the MNIST hand that has been uploaded before.

Introduction:

Adam is a replacement optimization algorithm for learning the stochastic gradient decent model of deep learning. Adam is relatively easy to configure where the default configuration parameters work best in most cases. A repeatable method (often abbreviated SGD) for stochastic gradient descent is to optimize an objective function (e.g., differential or subdivision) with appropriate polishing properties. The AdamaGrad and RMSProp algorithms combine the best features of Adam to provide an optimization algorithm for handling spar gradients in noise problems. It can be considered as a stochastic estimate of gradient decent optimization, since it replaces the actual one calculated by an estimate of the gradient (calculated from the complete data set). Data from randomly selected subsets). The problem with high-dimensional optimization in particular is that it reduces computational burdens, gaining rapid repetition in trades for low convergence rates. Root Mean Squad Propagation, or RMSProp, is an extension of gradient descent and an adgraded version of gradient descent that uses a partial gradient corrosion mean in step size adaptation for each parameter.

Results: For the results , I used ADAM optimizer and got 98.40% accuracy and loss 5.5%.



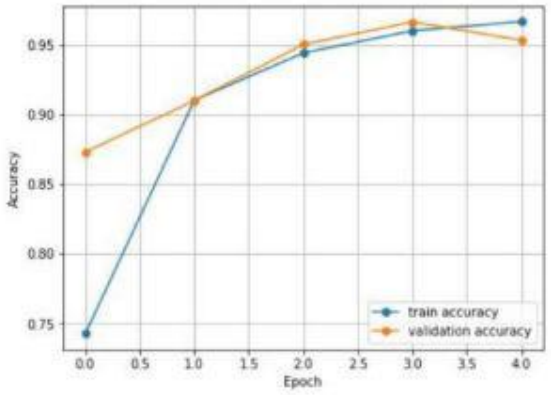


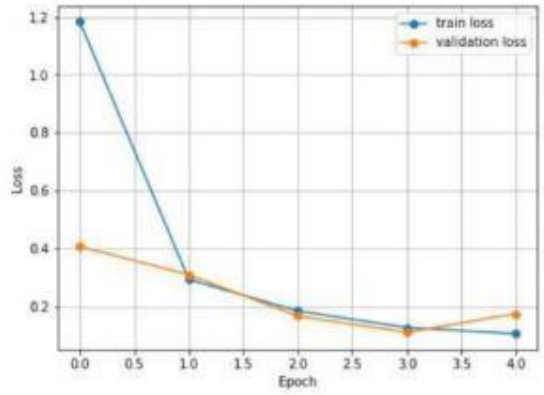

```
test_loss, test_acc = model_1.evaluate(X_test, Y_test)
print('\nTest Loss:', test_loss)
print('\nTest Accuracy:', test_acc)
```

```
313/313 [==============================] - 3s 9ms/step - loss: 0.0555 - accuracy: 0.9840

Test Loss: 0.05552656203508377
```

Then I Used SGD Optimizer with 95.37% accuracy.





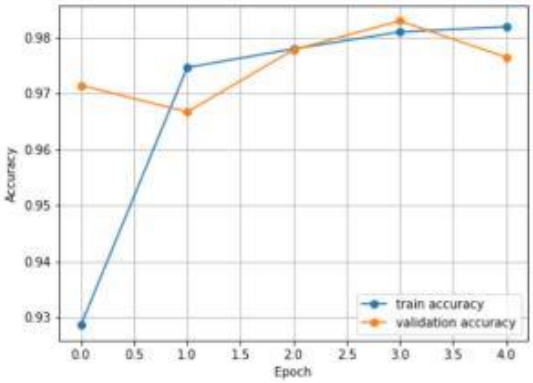

```
test_loss, test_acc = model_2.evaluate(X_test, Y_test)
print('\nTest Loss:', test_loss)
print('\nTest Accuracy:', test_acc)
```

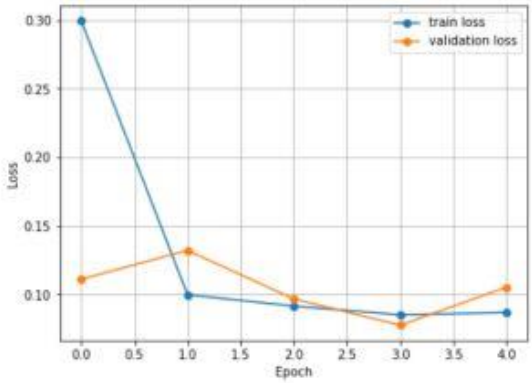
```
313/313 [==============================] - 1s 4ms/step - loss: 0.1590 - accuracy: 0.9537

Test Loss: 0.15898685157299042

Test Accuracy: 0.9537000060081482
```

At last RGBProp and the accuracy is 97.70%.






```
test_loss, test_acc = model_3.evaluate(X_test, Y_test)
print('\nTest Loss:', test_loss)
print('\nTest Accuracy:', test_acc)
```

```
313/313 [==============================] - 3s 9ms/step - loss: 0.0964 - accuracy: 0.9770

Test Loss: 0.09640488028526306

Test Accuracy: 0.9769999980926514
```

Discussion:

ADAM, SGD AND RMSProp for optimization of the EMNIST validation set has been used in this project. I am achieving 95.37% accuracy with SGD. When testing the same exact configuration withRMSProp the accuracy 97.70% and 98.40%. At first I faced some problem. Then I rerun the project and got the results.

References:

1. https://machinelearningmastery.com/adam-optimizationalgorithm-for□deep-learning/\

2. https://ruder.io/optimizing-gradient-descent/

3. https://medium.com/analytics-vidhya/a-complete-guide-to-adamand□rmsprop-optimizer□75f4502d83b

4.https://r.search.yahoo.com/_ylt=Awr9DtahLH1h1zAAFBVXNyoA;_ylu=Y29sbwNncTEEcG9zAzEEdnRpZAMEc2VjA3Nj/RV=2/RE=1635622177/RO=10/RU=http%3a%2f%2fen.wikipedia.org%2fwiki%2fConvolutional_neural_network/RK=2/RS=1WThgzdq_jM_g1lmFU1mMA658