Abstract

This report is about analysis of hand-written digits for training an OCR machine. Convolutional Neural Networks (CNN) Model was deployed and its parameters such as number of blocks, rate of learning and number of filters were varied accordingly. This model performance metrics showed that the model performed well with all set parameters except for that which had a learning rate of 0.1. Major change in parameters such as increasing the Convolution blocks from 3 to 4, augmenting the data and using L1 and L2 Regularization all gave good accuracy score and no overfitting occurred.

Introduction

CNN Model have been used for their ability to learn relevant features from raw pixel values. In this study, we aim to use CNN to correctly classify the handwritten digits in the MNIST dataset (LeCun, n.d.). The dataset consists of 70,000 images of handwritten digits, each of which is 28 by 28 pixels in size. This report is based on applying varying parameters with varying number of blocks in order to build the characterization model that will easily recognize handwritten

Methodology

The following steps were undertaken for this task:

1. **Data Preprocessing**:
   - **a.** The images' shape were converted from 3D to 4D.
   - **b.** The pixel values where normalized by dividing through by 255
2. **Building the Model**:

   The Model was built using several block of convolution with the following parameters.

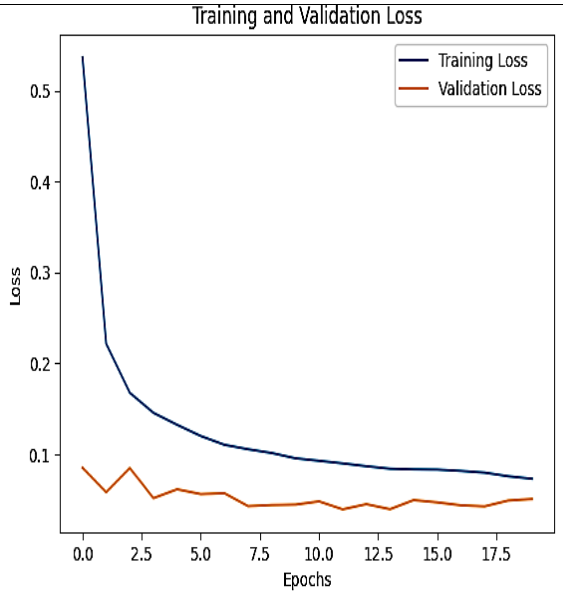
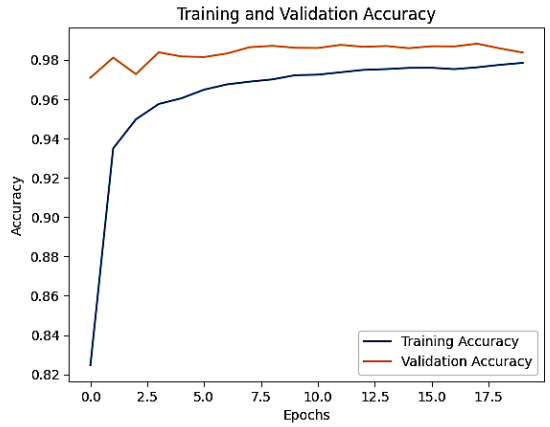
   i. Creating layers with specifics: Filter size, activation, padding, pool size.
   ii. Compiling the model: Using Adam Compiler, and varying the number of neurons, drop out value.
   iii. Training the model
3. Evaluation of the Model was then done using accuracy and loss curve, F1 Score and confusion matrix.

1.0. Results and Conclusion

QUESTION 1: HOW DID THE USE OF DIFFERENT REGULARIZATION METHODS AFFECT THE PERFORMANCE OF YOUR CNN MODEL?

USING VARIOUS REGULARIZATION METHODS

Table 1

MODEL 1: USING DATA AUGMENTATION		
	Performance Visualization	**Comments**
Training and Validation Loss		The decline of the training loss down below 0.2 shows that the model did well, also validation was seen to be below 0.1 indicating that there is no overfitting, the model generalized well.
Training and Validation Accuracy		The accuracy indeed was high and validation curve shows that the model generalized well
Classification Report	precision recall f1-score support 0 0.99 0.99 0.99 980 1 0.99 0.99 0.99 1135 2 0.96 0.97 0.97 1032 3 0.99 1.00 0.99 1010 4 0.99 1.00 0.99 982 5 0.97 0.97 0.97 892 6 0.97 0.97 0.97 958 7 0.99 0.99 0.99 1028 8 0.97 1.00 0.98 974 9 1.00 0.96 0.98 1009 accuracy 0.98 10000 macro avg 0.98 0.98 0.98 10000 weighted avg 0.98 0.98 0.98 10000	From the classification report, the F1 score of 0.98 showed that the model did well in classifying the images properly

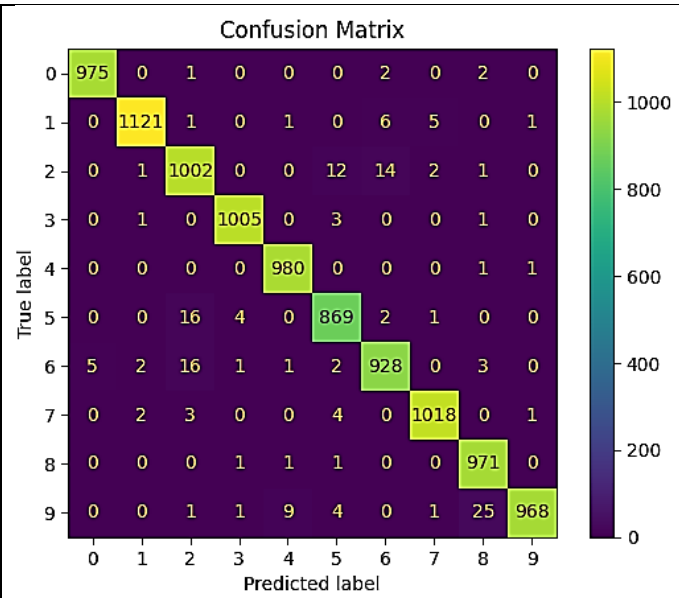
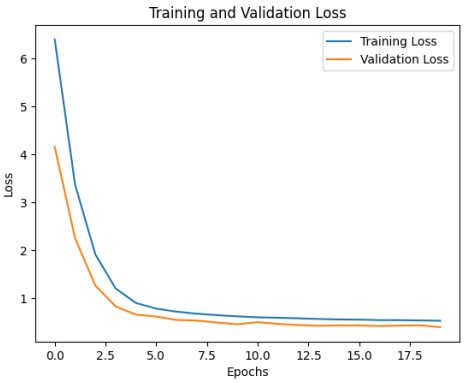
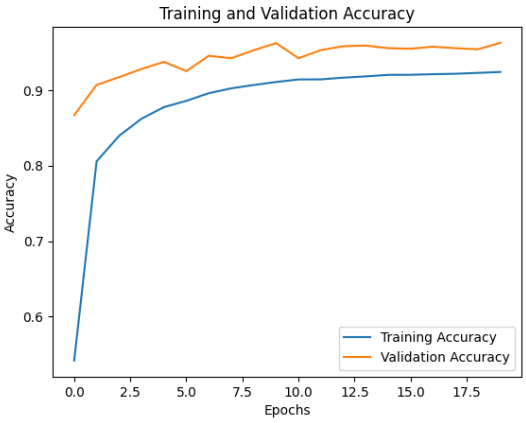
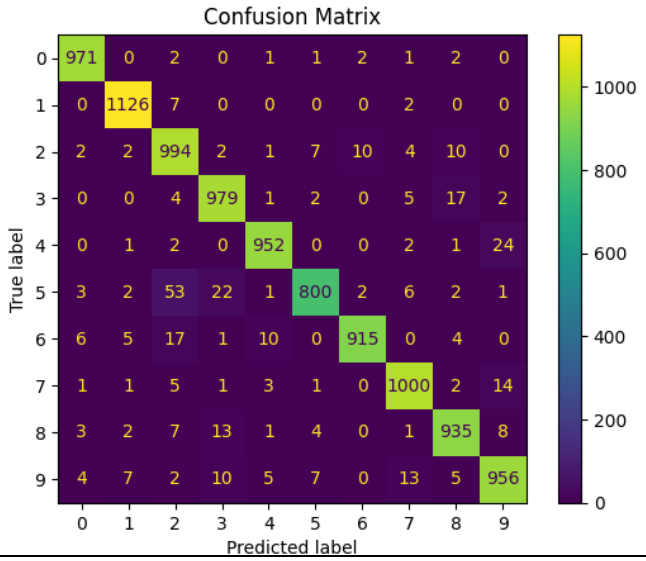
Confusion Matrix		The confusion matrix show that there far less miscategorization of the images, showing a good performance of the model.

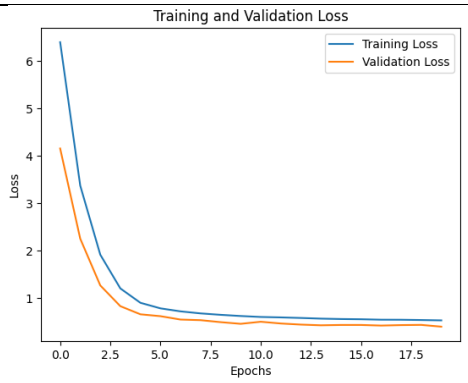
Table 2

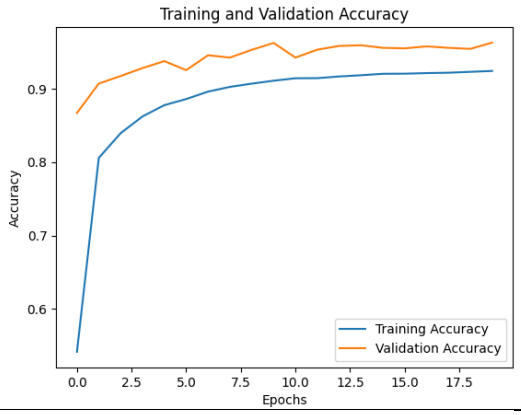
MODEL 2: USING L1 & L2 REGULARIZATION		
	Performance Visualization	**Comments**
Training and Validation Loss		The training loss curve shows a decline down below 1 at the 3rd epoch. This shows that the model did well, also validation loss curve was seen to be below that of training loss, indicating that there is no overfitting.
Training and Validation Accuracy		The accuracy indeed was high and validation curve is well above the training curve showing that the model did not over fit.

Classification Report		Compared to the previous model, the F1 score is lesser, however 0.96 as F1 score shows a good model.
Confusion Matrix		The confusion matrix show thatmost were categorized well.

	precision	recall	f1-score	support
0	0.98	0.99	0.99	980
1	0.98	0.99	0.99	1135
2	0.91	0.96	0.94	1032
3	0.95	0.97	0.96	1010
4	0.98	0.97	0.97	982
5	0.97	0.90	0.93	892
6	0.98	0.96	0.97	958
7	0.97	0.97	0.97	1028
8	0.96	0.96	0.96	974
9	0.95	0.95	0.95	1009
accuracy			0.96	10000
macro avg	0.96	0.96	0.96	10000
weighted avg	0.96	0.96	0.96	10000

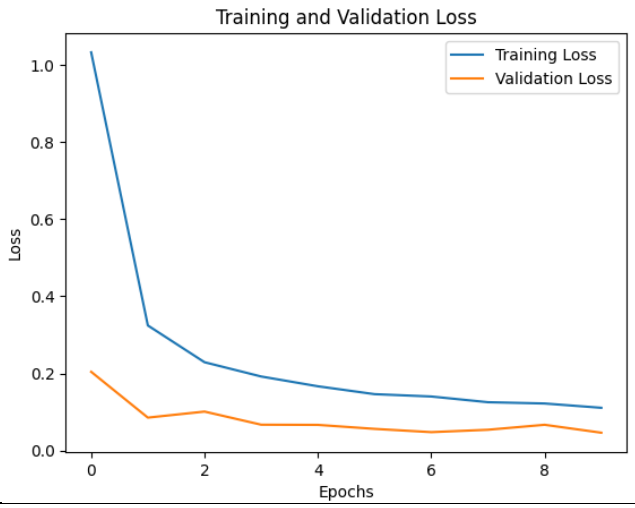
Table 3

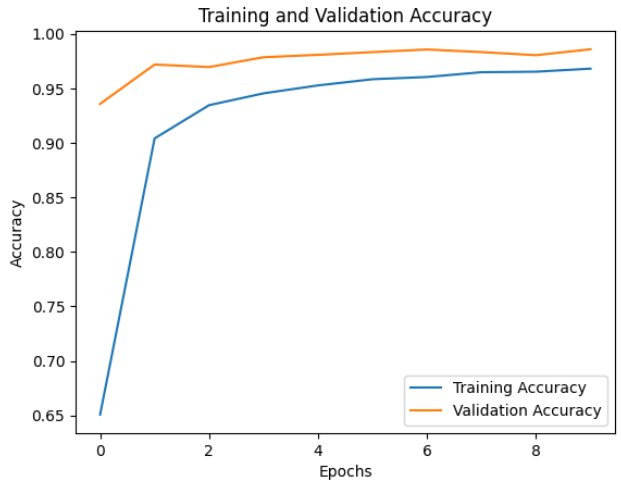
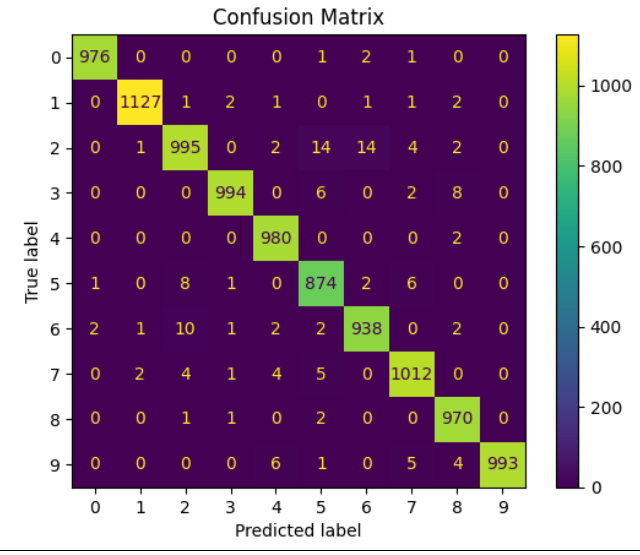
MODEL 3: USING EARLY STOPPING		
	Performance Visualization	**Comments**
Training and Validation Loss		From the application early stopping with a patience value of 10, the visualization fo the perfromance shows that it did not adversely affect the model as the model did just as well as the previous model

Training and Validation Accuracy		Validation loss curve still remained above the training curve, and the model did not overfit
Classification Report	precision recall f1-score support 0 0.99 0.99 0.99 980 1 0.99 1.00 0.99 1135 2 0.98 0.98 0.98 1032 3 0.98 0.98 0.98 1010 4 0.98 0.99 0.98 982 5 0.98 0.98 0.98 892 6 0.99 0.98 0.98 958 7 0.99 0.98 0.99 1028 8 0.98 0.98 0.98 974 9 0.98 0.97 0.98 1009 micro avg 0.98 0.98 0.98 10000 macro avg 0.98 0.98 0.98 10000 weighted avg 0.98 0.98 0.98 10000 samples avg 0.98 0.98 0.98 10000	From the classification report, the F1 score of 0.98 makes this model do just as well as the first model

QUESTION 2: REPORT HOW CHANGES TO THE NUMBER OF CONVOLUTION BLOCKS AFFECT THE PERFORMANCE OF YOUR MODEL QUANTITATIVELY?

Table 4

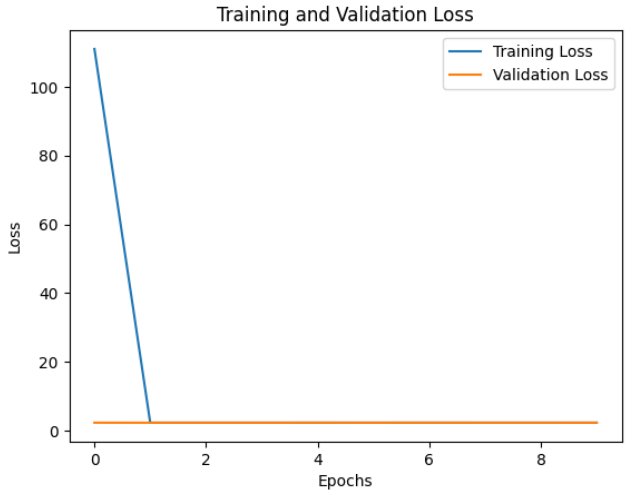
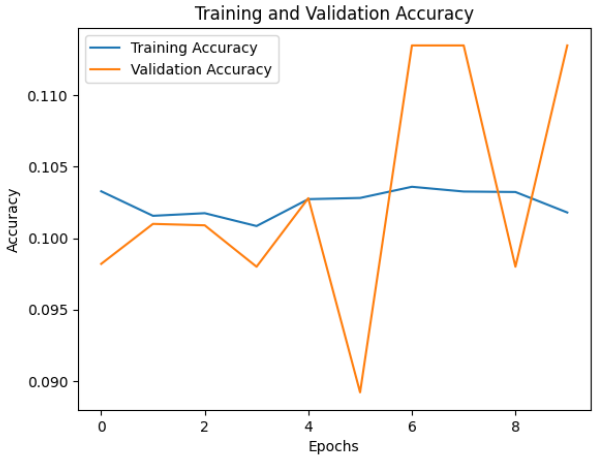
CHANGING THE NUMBER OF CONVOLUTION BLOCKS FROM 3 TO 4		
	Performance Visualization	**Comments**
Training and Validation Loss		The loss

Training and Validation Accuracy		The accuracy indeed was high and validation curve shows that the model generalized well
Classification Report		From the classification report, the F1 score of 0.98 showed that the model did well in classifying the images properly
Confusion Matrix		The confusion matrix show that there far less miscategorization of the images, showing a good performance of the model.

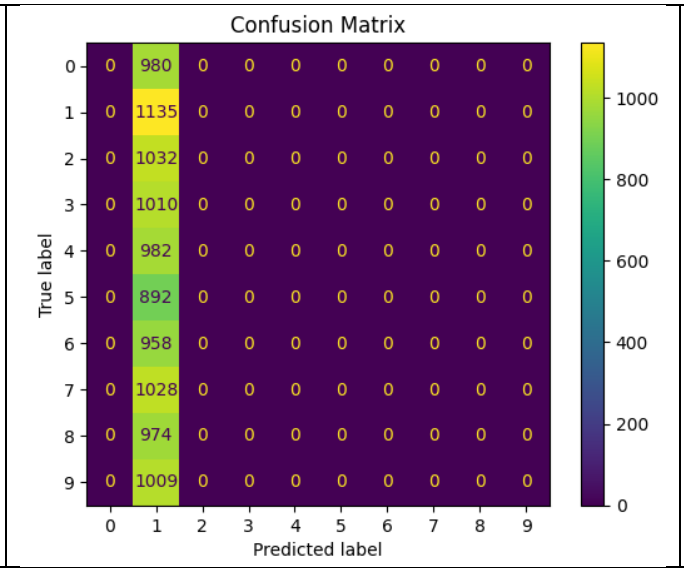
	precision	recall	f1-score	support
0	1.00	1.00	1.00	980
1	1.00	0.99	0.99	1135
2	0.98	0.96	0.97	1032
3	0.99	0.98	0.99	1010
4	0.98	1.00	0.99	982
5	0.97	0.98	0.97	892
6	0.98	0.98	0.98	958
7	0.98	0.98	0.98	1028
8	0.98	1.00	0.99	974
9	1.00	0.98	0.99	1009
accuracy			0.99	10000
macro avg	0.99	0.99	0.99	10000
weighted avg	0.99	0.99	0.99	10000

QUESTION 3: WHAT IS THE EFFECT OF VARYING LEARNING RATES ON THE PERFORMANCE OF THE CNN ALGORITHM

Table 5

CHANGING THE CUMBER OF CONVOLUTION BLOCKS TO FOUR		
	Performance Visualization	**Comments**
Training and Validation Loss		A steep decline of the Training loss was achieved, however it stopped epoch 1 and merged with the validation loss which remaind uniform at all epochs. This showed that the learning rate is not appropriate for the model to generalize well, rather it caused overfitting
Training and Validation Accuracy		Very low accuracy and erratic validation accuracy curve indicates that learning rate of 0.1 cannot be recommended. At Epoch of 6 the model overfitted greatly, dropped sharply at 8 and still overfitted at 9. In all the model did not do well.
Classification Report		From the classification report, the F1 score of 0.1 proving further the failure of 0.1 learning rate

	precision	recall	f1-score	support
0	0.00	0.00	0.00	980
1	0.11	1.00	0.20	1135
2	0.00	0.00	0.00	1032
3	0.00	0.00	0.00	1010
4	0.00	0.00	0.00	982
5	0.00	0.00	0.00	892
6	0.00	0.00	0.00	958
7	0.00	0.00	0.00	1028
8	0.00	0.00	0.00	974
9	0.00	0.00	0.00	1009
accuracy			0.11	10000
macro avg	0.01	0.10	0.02	10000
weighted avg	0.01	0.11	0.02	10000

Confusion Matrix		The confusion matrix showed the miscategorization of the image with learning rate of 0.1

QUESTION 4: WAS THERE A CASE OF OVERFITTING OBSERVED IN YOUR MODEL AT ANY POINT ? EXPLAIN

There was no overfitting with the following parameters of the model

1. Using three and four convolution blocks,

2. Apply early stopping with patience value of 10,

3. Using L1 and L2 Regularization, and

4. Augmenting the data.

All the above did not lead to overfitting, as evidently see in the loss and accuracy curve plotted in Table 1, 2, 3 and 4 above, all validation curve in these plot were above the training curve.

However, with a learning rate of 0.1, overfitting occurred as shown in the Training and Validation Accuracy in Table 5.

This can be seen from the trend of various changes in parameter of the model. The validation curve in the plot were seen to be higher that the training curve.

CONCLUSION

In conclusion, this study demonstrated the effectiveness of using Convolutional Neural Networks (CNN) for image classification. Optimal performance were seen to be at 3 convolution blocks, also it was seen to improve when it was increased to 4 blocks and at a learning rate of 0.001. With this, it can be seen how tuning the hyperparameters led to good performance of the model.

REFERENCES

Yann LeCun, n.d.*THE MNIST DATABASE of handwritten digits*, n.d. Retrieved from: http://yann.lecun.com/exdb/mnist/ Accessed: 25th May 2023.