Multi-Modal framework for classification of handwritten digits using Images and Audio

Neha Joshi
Department of Electircal and Computer Engineering
Texas A&M University
College Station, TX 77845
nehayj100@tamu.edu

Abstract

A multi-modal learning technique was proposed to classify handwritten digits including two data modalities - Image and Audio. Handwritten image data from MNIST and corresponding spoken audio samples were considered as a part of database. A multi-modal deep learning network was trained using a 4 layer Convolutional Neural Network (CNN) for images and a 1 layer Recurrent Neural Network (RNN) for audio. Latent fusion technique was used to combine these 2 modalities. The accuracy of the fused model was higher than that of the image and audio models individually. An accuracy of 99% was achieved in the multi-modal model which outperformed the individual model results.

1 Introduction

Handwritten digit classification is a very well known problem and have been traditionally solved using image data. MNIST is a well known dataset often used for this task. It was realized that handwriting, English digit styles and fonts can vary immensely as per geographies and hence classification accuracy can go down if any other images outside MNIST are fed to the model. It might be valuable to add another modality to the modal which might help to preserve interpretability of the information. Audio is an excellent example which can pair up with images seamlessly to crease the model. We thus, propose a model that is trained by the information of digits from images as well as audio on separate planes and then fused.

[1] provides an excellent literature review of how multi-modal techniques evolved and can be used in various use cases right from image , audio to text classification and combination of these. [2] explains the use of multi-modal model in the classification of images including text as the second modality for graphic information classification. Similarly [3] uses text and images for document classification application. We also see references of use of audio as a modality along with images in [4] where acoustic images are classified using audio visual representations; [5] have used audio and images for speech recognition; Video classification has also been done successfully by [6]. Looking at the promising results of these analyses, a potential was seen in improving results of image classification using multi-modal deep learning with the use of audio in conjunction.

An accuracy of 99% was achieved using the two representations which was above most of the MNIST benchmarks. Individual models performed well with accuracies 98.6 % and 93% respectively for image CNN and audio RNN but the multi-model model outperformed these individual models. Rigorous hyper parameter tuning was performed to get optimized parameters for all the three models (image, audio and multi-modal).

2 Your Method

The method used is designed in 3 steps:

1. Image Model: We build a Convolutional Neural Network based image model with 4 CNN layers and 3 linear layers at the end. We added Batch Normalization and a drop out after each linear layer. The image model returns two parameters- final layer (10 class nodes) and embedding of the layer prior to the final layer.

2. Audio Model: We build a Recurrent Neural Network for the audio model as audio is a sequence data which means the relation between consecutive data points is important. The network was tuned and consisted of 1 RNN layer and 3 linear layer following that. We also applied batch normalization to each layer without any dropout.

3. Multi-modal model with latent fusion: The image and audio models described above were fused together to create a multi-modal model. Early and latent fusions were 2 options for fusion. Latent fusion was selected to make the model more intuitive in terms of contributions from each individual model as well as the multi-modal model.

The report is structures as follow: We present the data pre-processing steps and details of how both the data modalities are loaded. A detailed explanation of the model follows which describes exact architecture of the used CNNs, RNNs and fully connected neural network layers. Further the model training and hyper-parameter tuning is explained. Finally we present our results and conclude the work.

2.1 Data Preprocessing

The data is available on the course Kaggle page here The dataset is a multimodal MNIST dataset. It contains images of written digits as well as the audio of spoken digits, paired with the corresponding label (0-9). As a first step, all data files (iamge training, image testing, audio training, audio testing) were imported adn converted into tensors. Image Data: The images were reshaped to 28x28 and used for the CNN image model. We had total 60,000 images of handwritten digits in the training set and 10,000 images in the testing set. The training data was spitted into training validation set in the ratio of 80%-20% respectively. For each of these datasets (train, test and validation), we create data loaders to get data in batches of 64 each. Thus, effectively we get 750 batches in the training data, with last batch having lesser examples. We finally plot a few images and audio samples from first batch to visualize how the data looks.

Figure 1: Images of handwritten digits in batch 1

Audio Data: Audio data is a sequence of 507 points that form the audio. We have 60,000 audio samples in training data and 10,000 audio samples in testing set. Out of the 60,000 training samples, we split it into training validation sets in ratio 80% - 20% respectively. There are two ways to analyse the audio data- one is to use it as a 1D sequence and other is to convert it into spectograms (frequency-time images) and consider it as an image classification problem. We select the prior method and simply use the audio data as a 1D sequence.

2.2 Model Design

The model is designed in 3 major steps -Image model, Audio model, combine embeddings of the image and audio model to fuse into a multi modal model. Fig 3 shows an overall block diagram of how the model has been designed and executed.

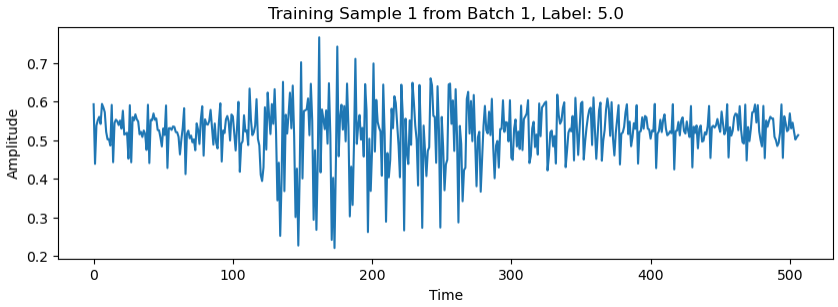


Figure 2: Audio of digits

1. Image Model Architecture:
We use four 2D convolutional layers of the dimensions: 6, 12, 24 and 48 respectively. Post this we apply three fully connected layers of size 120, 64, 10 respectively. Batch normalization and drop-out of rate 5% is applied after each linear fully connected layer. Embeddings from the second last layer of size 64 are pulled and are passed as image features to the multi-modal model. These emebeddings are of size [batch size = 64, nodes in layer= 64]. Hyper parameter tuning was performed for the parameters, layers to apply batch normalization, filter size and activation function. More details on tuning can be found in the following sections. Other parameters that were not tuned were taken by analysis with respect to accuracy and required model performance.

2. Audio Model Architecture: Audio being a time series sequence, we use a Recurrent Neural Network to classify the audio signal. We use an input layer size of 507 as each audio signal has 507 points. Further we use a hidden size of 320 and 1 RNN layer (these 2 are tune). Further 3 linear layers are used of sizes 100, 64 and 10 with batch normalization in each step but no drop-out. We pull the embedding of audio model from the second last layer of size 64. SO finally we have the audio embedding of size [batch size = 64, nodes in layer= 64].

3. Multi-modal latent fusion model: As shown in Fig. 3, the embeddings of size [64,64] from both image and audio model are fused (latent fusion as we fused the embeddings after being processed from individual models) and passed to a multi-modal model with architecture of 1 fully connect layer of size input 64+64 and output 10. Finally an argmax is taken to classify the inputted (image+audio) set into the corresponding labels. From each of the image and audio models, we return the final layer outputs as well as embeddings so that they can be passed on further. // We had also tried the multi-modal model with embedding size of 10 and it preformed well but size 64 outperformed over other architecture tried so it was finalized.

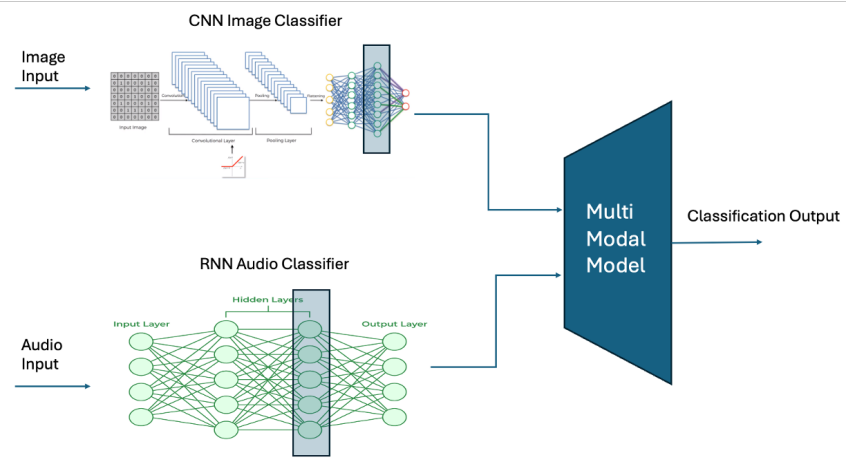


Figure 3: Overall model design

2.3 Model Training

Similar training algorithms were followed for all the three models with slight change in number of epochs as per the model requirement. Image and Audio model Training: The models were initially trained for 10,15 and 20 epochs and it was found that the CNN for image model trained pretty well in 10 epochs (achieved a validation accuracy of 98. The audio RNN model took 25 epochs to get trained upto an accuracy of 93%. The training process for RNN for audio was seemingly slower than that of CNN for images. CrossEntropyLoss was used for both the models with Adam optimizer. Learning rate for CNN was 0.01 while for RNN 0.0001 learning rate was used. Various rates like 0.1, 0.01, 0.001, 0.0001 were tried and the most optimum rate for both the models was picked. Form the last epoch of last batch of the best model(chosen by hyperparamter tuning described in next section), we pull out the embeddings of the second last layer and save them. We parallelly also keep collecting all batch y labels and batch wise into a list so as to use them for further analysis related to dimensionality reduction and clustering analysis. As per the pre-defined process for training a neural network- we first perform back-propagation to set parameters and the stop the backpropagaton to test the model on validation data. For each epoch and batch, we calculate the loss, training and validation accuracies.

2.4 Hyperparameter Tuning

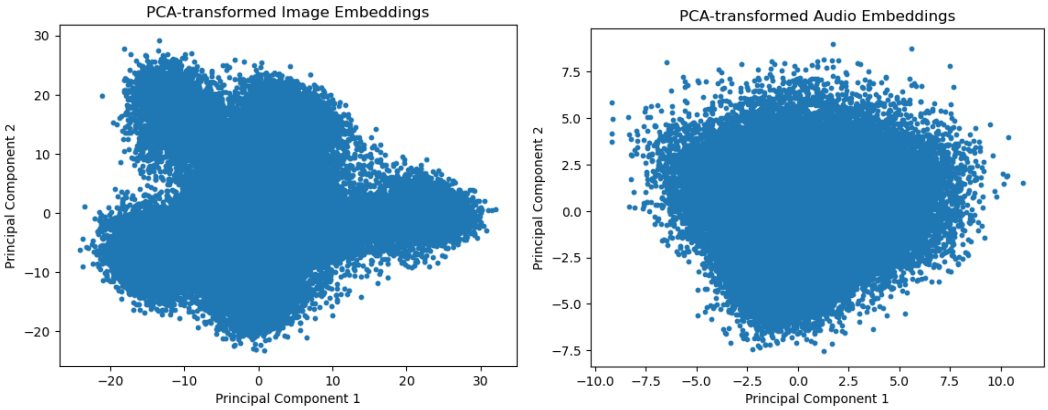
For each of the models, crucial parameters were tuned in the following way:
1. Image model: For the image model, three parameters were optimized- Layers to be batch normalized (three options were explored- ["l1","l2","l3","l4","l5"], ["l1","l2"],["l1","l3","l4"] ; filter size - 3x3 and 5x5 ; Activation functions used - ["tanh", "relu"].

After comparing the validation accuracy's, it was observed that the best parameters were- normalize all layers (["l1","l2","l3","l4","l5"]), filter size of 3x3 and tanh activation function were best.
2. Audio model: For the audio RNN model the hyper-parameters tuned were- hidden layer size - tried the options [320,350,400] and number of hidden layers- [1,2].It was found that 320 hidden size and 1 RNN layer were most optimum to reach the maximum validation accuracy.
3. Multi-modal model: The number of fully connected layers were treated and hyper parameters and two models were tested. It was found that having a single fully connected layer was most optimum.

Hence, a rigorous hyper-parameter tuning has been performed by creating a total of 20 models and best mdoel combinations were used.

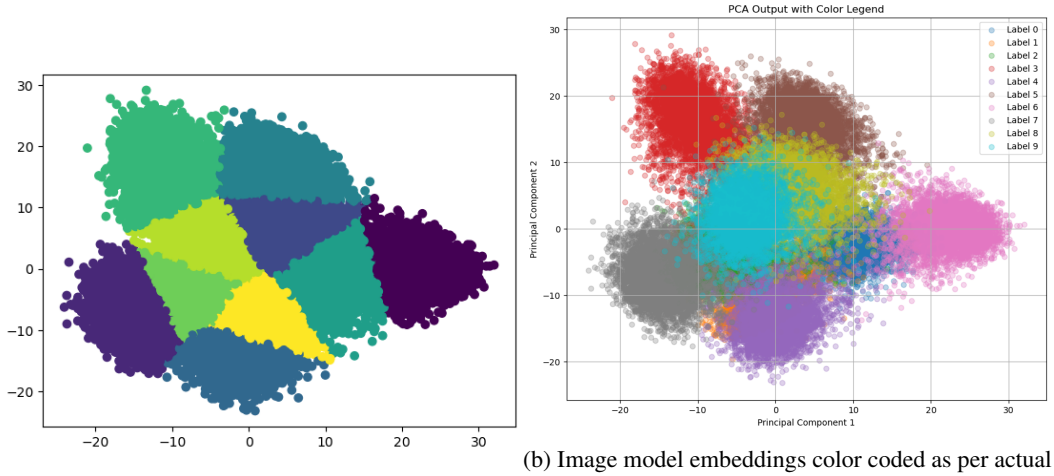
3 Dimensionality Reduction and Clustering Analysis

We use the Image and Audio embeddings separately and perform dimentionality reduction using Principle Component Analysis on each of the embeddings. As mentioned in the training process, we have pulled the embeddings for all batches in a list that are inputted into the PCA [7], [8]. Further we have also carried on a list with labels of the corresponding embeddings. We first plot the PCA and perform K-means clustering with 10 clusters (separately for image and audio embeddings)- This can be seen in Fig. 5a and Fig. 6a. We go ahead and now plot the embeddigns in Fig 5b and Fig 6b and color code the points with the corresponding pulled actual labels. We them compare Fig.5a with Fig.5b and Fig 6a with Fig 6b to see the correspondance between actual clusters and those formed by K means. It was seen that the image model show very evident clusters and both the plots are comparable but the audio model did not show very clear 10 clusters in the embedding when coded with actual labels. This shows that the image model is much stronger and has higher contribution in the multi-modal model.



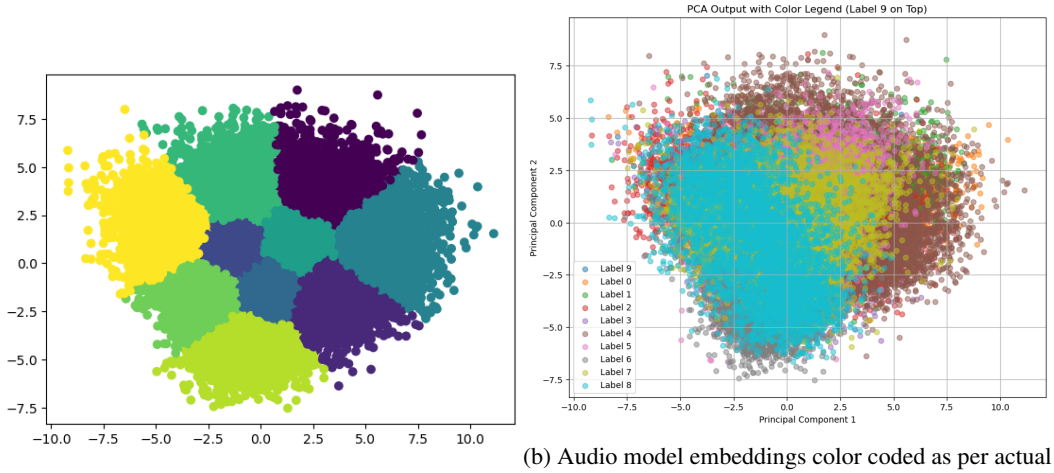
(a) Image model embeddings (b) Audio model embeddings

Figure 4: Model embeddings



(a) Image model embeddings clustered using K means labels (b) Image model embeddings color coded as per actual

Figure 5: Comparison of image model embeddings



(a) Audio model embeddings clustered using K means labels (b) Audio model embeddings color coded as per actual

Figure 6: Comparison of audio model embeddings

4 Results

The proposed Image, Audio and multi-modal models were tested extesively for multiple hyper-parameters and the Fig. 4 and Fig 5 show the results for accuracies across epochs for hyper-parameter tuning of Image and Audio models respectively.

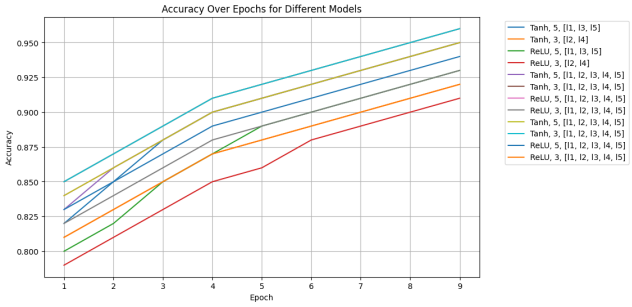


Figure 7: Hyper-parameter Tuning for Image Model

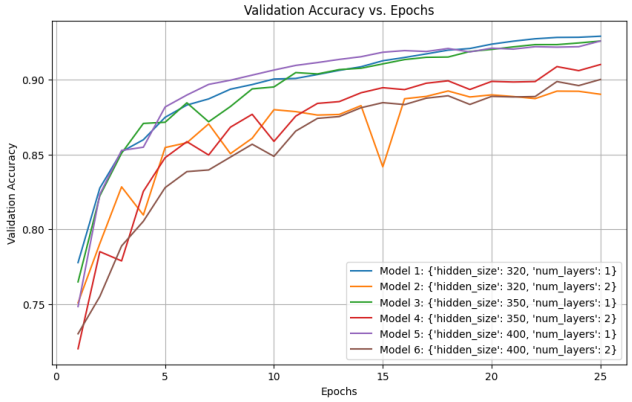


Figure 8: Hyper-parameter Tuning for Audio Model

The best models were found out to be:
1. Image model: 4 Convolutional layers, 3 linear layers, all 3 linear layers batch normalized and drop-out applied. Tanh activation and filter size of 3x3.
2. Audio Model: 1 RNN block with hidden size of 320 followed by 3 linear layers were the most optimum. 25 epochs needed to train model upto 93% validation accuracy.
3. Multi-modal model: 1 linear layer for the fused embeddings was optimum.

We finally get an accuracy of 99% for the multimodal model which is promising for the use-case. All the model accuracies are tabulated in Table 1.

Final Model	**Accuracy**
Image CNN	98.48%
Audio RNN	92.9%
Multimodal Model	99.0%

Table 1: Accuracy of Final Models

```
CNNModel(
  (conv1): Conv2d(1, 6, kernel_size=(3, 3), stride=(1, 1))
  (conv2): Conv2d(6, 12, kernel_size=(3, 3), stride=(1, 1))
  (conv3): Conv2d(12, 24, kernel_size=(3, 3), stride=(1, 1))
  (conv4): Conv2d(24, 48, kernel_size=(3, 3), stride=(1, 1))
  (fc1): Linear(in_features=768, out_features=120, bias=True)
  (fc2): Linear(in_features=120, out_features=64, bias=True)
  (fc3): Linear(in_features=64, out_features=10, bias=True)
  (batch_norm1): BatchNorm2d(6, eps=1e-05, momentum=0.1, affine=True, track_running_stats=True)
  (batch_norm2): BatchNorm2d(12, eps=1e-05, momentum=0.1, affine=True, track_running_stats=True)
  (batch_norm3): BatchNorm2d(24, eps=1e-05, momentum=0.1, affine=True, track_running_stats=True)
  (batch_norm4): BatchNorm2d(48, eps=1e-05, momentum=0.1, affine=True, track_running_stats=True)
  (dropout): Dropout(p=0.05, inplace=False)
)
```

Figure 9: Best Optimized CNN model for Image Classification

```
RNNAudioModel(
  (rnn): RNN(507, 320, batch_first=True)
  (fc1): Linear(in_features=320, out_features=100, bias=True)
  (bn1): BatchNorm1d(100, eps=1e-05, momentum=0.1, affine=True, track_running_stats=True)
  (fc2): Linear(in_features=100, out_features=64, bias=True)
  (bn2): BatchNorm1d(64, eps=1e-05, momentum=0.1, affine=True, track_running_stats=True)
  (fc3): Linear(in_features=64, out_features=10, bias=True)
  (bn3): BatchNorm1d(10, eps=1e-05, momentum=0.1, affine=True, track_running_stats=True)
)
```

Figure 10: Best Optimized RNN model for Audio Classification

5 Conclusion

As per the achieved results, it could be seen that the Image model performs exceptionally well. The design decision in terms of the layers and parameters were very optimum. The audio model also reached a substantial accuracy but it was observed that the accuracy could be further increased. One important factor that can be considered is there are other methods that can be used to classify audio data like use of images produced by audio also known as spectograms and considering this as an image classification problem. Apart form that, 1 D convolutional layers can also be used in place of RNNs. Finally, early fusion can also be tested to check if results are improving.

References

[1] Skowron, A., Wang, H., Wojna, A. and Bazan, J., 2006. Multimodal classification: case studies. In Transactions on Rough Sets V (pp. 224-239). Springer Berlin Heidelberg.

[2] Kim, E. and McCoy, K.F., 2018, October. Multimodal deep learning using images and text for information graphic classification. In Proceedings of the 20th International ACM SIGACCESS Conference on Computers and Accessibility (pp. 143-148).

[3] Audebert, N., Herold, C., Slimani, K. and Vidal, C., 2020. Multimodal deep networks for text and image-based document classification. In Machine Learning and Knowledge Discovery in Databases: International Workshops of ECML PKDD 2019, Würzburg, Germany, September 16–20, 2019, Proceedings, Part I (pp. 427-443). Springer International Publishing.

[4] Perez, A., Sanguineti, V., Morerio, P. and Murino, V., 2020. Audio-visual model distillation using acoustic images. In Proceedings of the IEEE/CVF winter conference on applications of computer vision (pp. 2854-2863).

[5] Mroueh, Y., Marcheret, E. and Goel, V., 2015, April. Deep multimodal learning for audio-visual speech recognition. In 2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) (pp. 2130-2134). IEEE.

[6] Jiang, Y.G., Wu, Z., Tang, J., Li, Z., Xue, X. and Chang, S.F., 2018. Modeling multimodal clues in a hybrid deep learning framework for video classification. IEEE Transactions on Multimedia, 20(11), pp.3137-3147.

[7] CSCE 633 Discussion for K means clustering, Texas A&M University

[8] CSCE 633 Lecture slides, Texas A&M University