Homework 5

Mu-Ruei Tseng
Computer Science Department
Texas A&M University
College Station, TX 77845
mtseng@tamu.edu

Abstract

This project explores a multimodal framework for improving image and audio classification, particularly when the data is ambiguous or noisy. We utilize the Written and Spoken Digits Database (1), which contains images from the MNIST dataset with corresponding MFCC features from the one-second audio recordings of that digits. We experimented with three different models: Image-based (ImageNet), Audio-based (AudioNet), and a multimodal structure model (HybridNet) that integrates both modalities. Our results demonstrate that the multimodal model achieves higher accuracy (HybridNet_CNN: 99.74%, HybridNet_CNN_LSTM: 99.36%) than using either image (ImageNet_CNN: 99.13%) or audio (AudioNet_CNN: 97.75%, AudioNet_LSTM: 97.18%) features alone, suggesting that the use of both features enhances prediction by utilizing complementary features to clarify classifications when one modality is ambiguous.

1 Introduction

Image and audio classification tasks have been widely studied, with each field achieving significant accuracy over the years. For image classification, Convolutional Neural Networks (CNNs) are commonly used to extract local features and spatial hierarchies in image data, enabling the effective recognition of patterns, shapes, and objects within images. Conversely, audio data often utilize Recurrent Neural Network-based models such as LSTMs or Transformers to capture temporal information. Given their distinct information representations, one effective approach is to use a multimodal network that integrates both the image and audio data as inputs, leveraging the strengths of both modalities. This can be particularly advantageous when one of the data is more noisy and challenging to classify. Figure 1 displays samples from the MNIST dataset. We trained a simple CNN-based model that although achieving an accuracy over 99%, fails to correctly classify these particular samples due to their inherent ambiguity. However, if we provide additional information, such as the audio data, it may clarify such confusion. Therefore, in this paper, we explore various combinations of the input data, including using pure image, audio, and their combination. We designed three types of models: image-based (we called it ImageNet) audio-based (AudioNet), and the multimodal network (HybridNet). We aimed to show that using a multimodal structure enhances classification accuracy. We conduct experiments using the Written and Spoken Digits Dataset (1), details of which, along with our model design, will be discussed in the following sections.

2 Our Method

2.1 Data Preprocessing

In this paper, we use the Written and Spoken Digits Dataset (1) which contains two types of data: handwritten digits and spoken digits. Handwritten digits data consists of 70000 images (60000 for training and 10000 for test) of 28 x 28 = 784 dimensions, which is extracted from the MNIST dataset

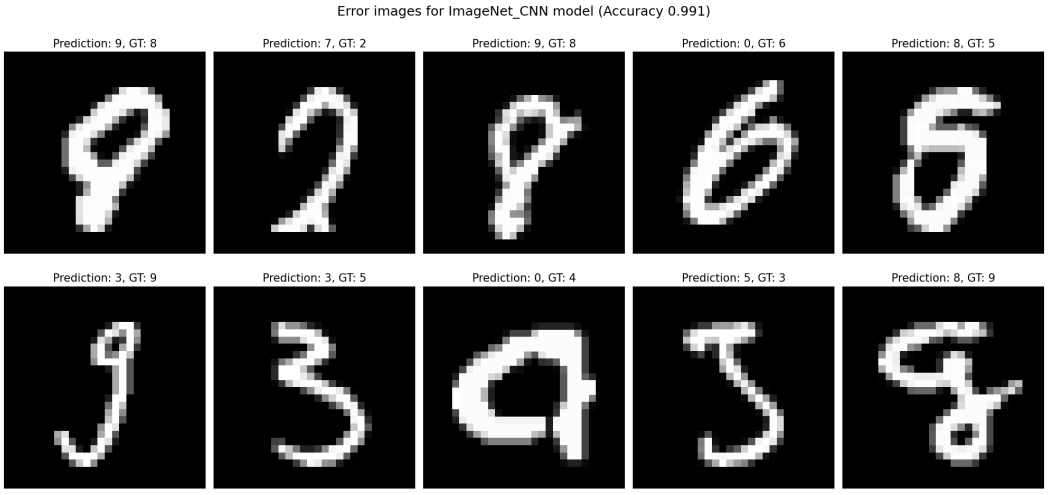


Figure 1: Here is the result using only the image as the input and trained on a CNN-based model. Although the model is achieving an accuracy of 99%, it is difficult for it to classify these samples.

with no additional processing. Spoken digits, on the other hand, contain 38908 utterances of the ten digits (34801 for training and 4107 for test). Since the number of spoken digit data is less than handwritten digit data, they duplicated some random samples to match the amount. Each audio sample is approximately one second long and they segment it into 39 time slots and find the 12 MFCC coefficients with an additional energy coefficient for each segment respectively. Therefore, for each sample, they have a final vector of 39 x 13 = 507 dimensions. They also applied standardization and normalization to the MFCC features. Sample data are shown in Figure 2 and 3. We performed data preprocessing for the image data by normalizing it with mean $\mu = 0.5$ and standard deviation $\sigma = 0.5$. Since the MFCC data provided in the dataset is already standardized and normalized, we do not perform any preprocessing for the audio data.

We split the data into train, validation, and test sets according to the ratio: 14:3:3. All the results' accuracy and f1 score are evaluated on the test set.

2.2 Model Design

In this section, we detail our various model types. There are three primary models: the Image-based model, the Audio-based model, and the Multimodal (Hybrid) model.

2.2.1 Image-based model

Since we are working with a relatively small and simple dataset (MNIST), we designed the model using a simple CNN network structure with fully connected layers at the end for classifying the data. See the detailed model implementation in Table 1.

2.2.2 Audio-based model

Since our input data is the MFCC features within a 39-segment time period, the audio data is in the shape of 39×13. We experimented with two different approaches: CNN-based (AudioNet_CNN) and RNN-based (AudioNet_LSTM). The CNN-based design, similar to our ImageNet_CNN model, treats the input as an image and finds the MFCC feature patterns across both local temporal and feature dimensions, effectively capturing spatial relationships within the time-frequency representation of the audio. The detailed model parameters and layers are described here 2. On the other hand, since audio data contains temporal information, it might be more informative if we use an RNN-based model that sequentially processes the time-dependent features, allowing for the extraction and integration of temporal dynamics inherent in the audio file across the entire audio sequence, which is crucial for understanding the audio. The model structure is defined here 3. For the choice of RNN-based model,

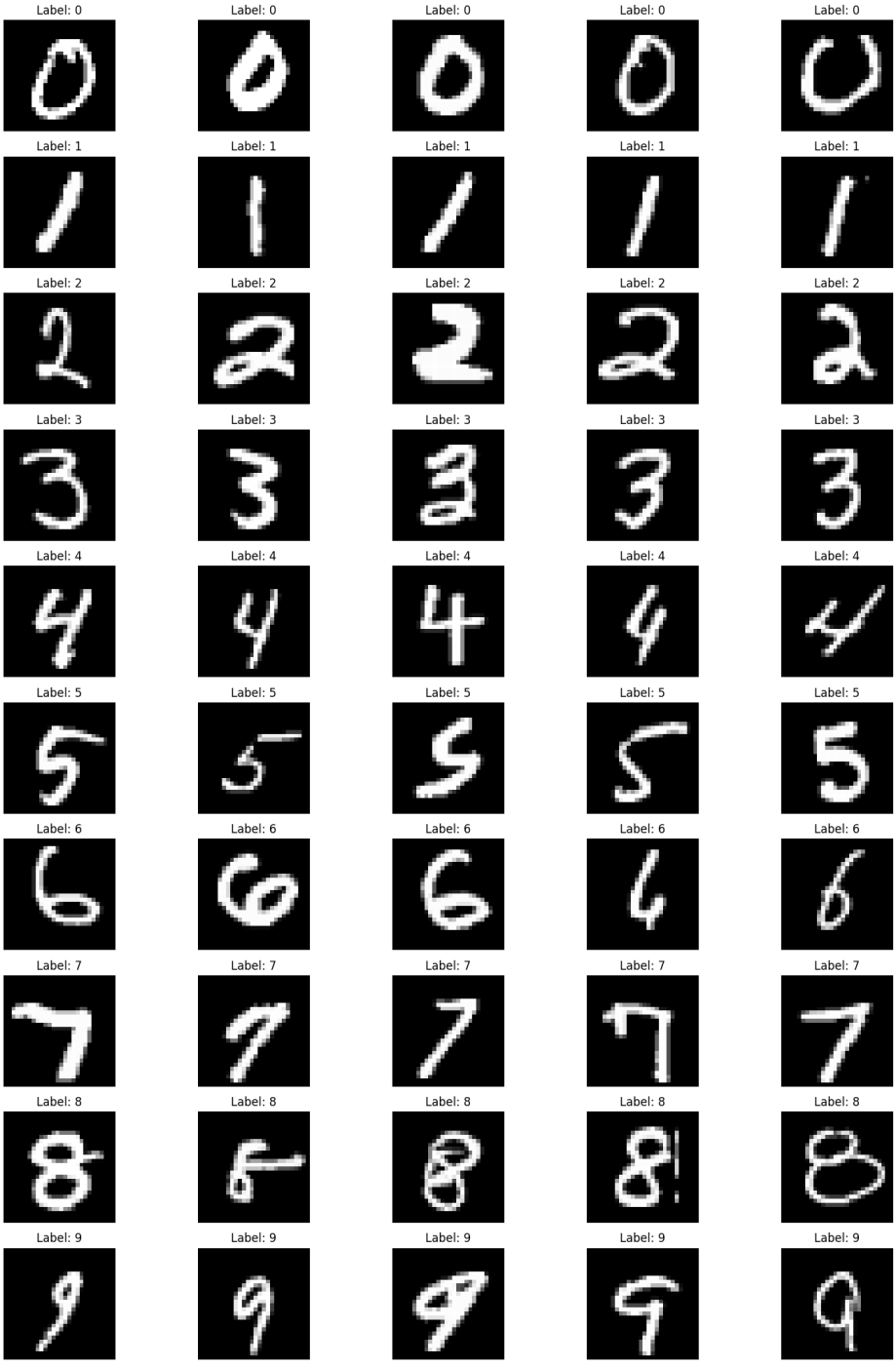


Figure 2: Image Data from the dataset (1).The image has dimensions of 28×28 pixels and is a grayscale image.

we use a unidirectional LSTM model as our audio sample only contains the speaking of a digit, which does not require backward contextual information for effective interpretation. The pronunciation of a digit is largely affected by the sound that came before, not what comes after. The model configuration is shown in 3

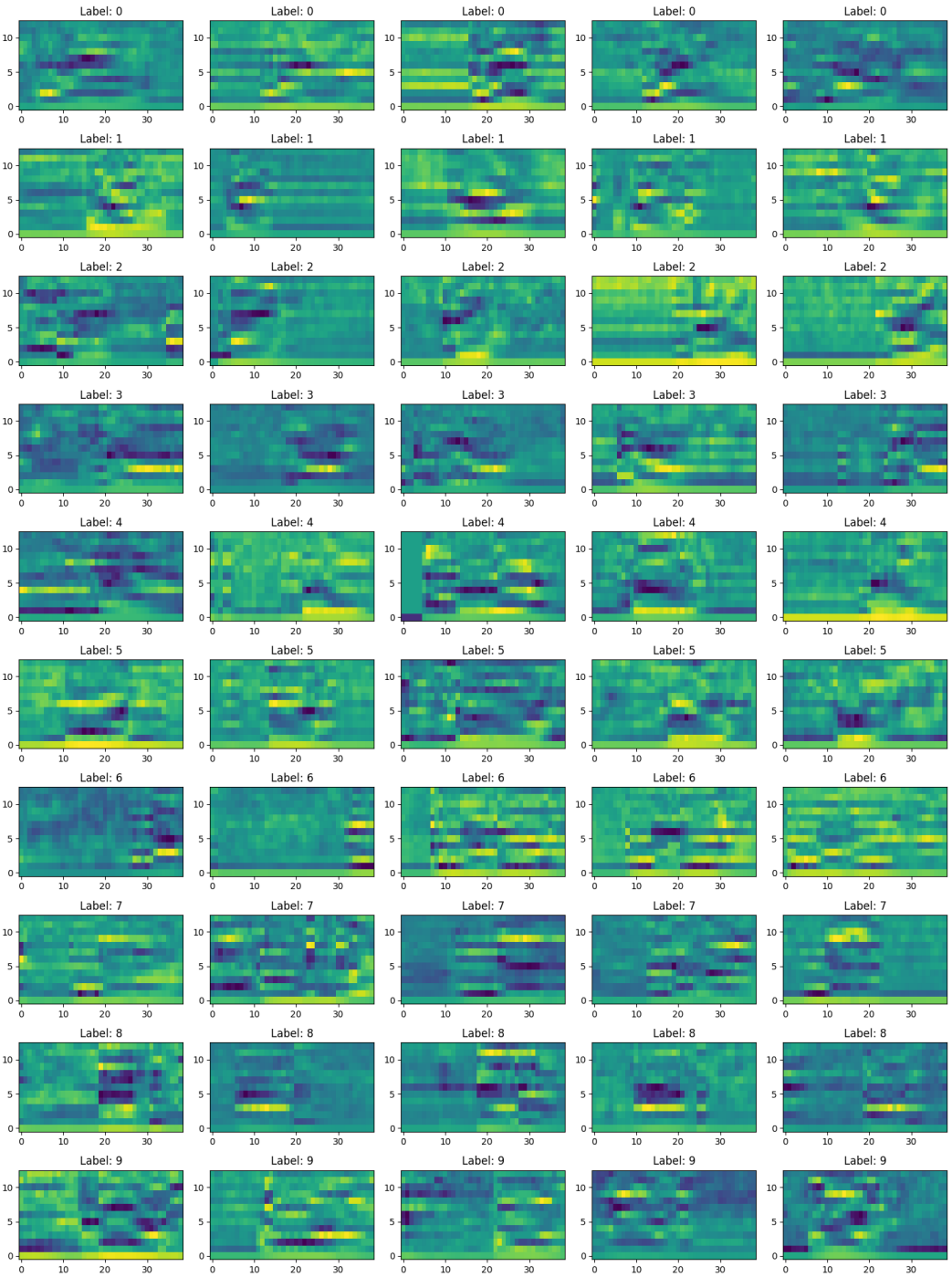


Figure 3: Audio Data from the dataset (1). It contains the 12 MFCC features with one additional energy coefficient for the one-second audio signal speaking that digit.

2.2.3 Multimodal (Hybrid) model

For the multimodal model, we employ the same encoder structures as those used in the Image-based and Audio-based models. Since we experimented with both CNN-based and RNN-based approaches for the audio model, we have developed two versions of the multimodal model as well. We named the models as HybridNet_CNN and HybridNet_CNN_LSTM. The model encoder details are shown

Table 1: Image-Based Model (ImageNet_CNN). Each ConvBlock includes a BatchNorm2d layer followed by ReLU activation.

Hidden Layers	Output Size	Operations	# of Filters
ConvBlock	28×28	3x3 convolution, padding=1, stride=1	32
Pooling	14×14	2x2 max pooling, stride=2	-
ConvBlock	14×14	3x3 convolution, padding=1, stride=1	64
Pooling	7×7	2x2 max pooling, stride=2	-
Dropout	-	Dropout with $p = 0.25$	-
Flatten	3136	-	-
Fully Connected layer	128	ReLU activation	128
Dropout	-	Dropout with $p = 0.5$	-
Fully Connected layer	10	-	10

Table 2: Audio-Based Model (AudioNet_CNN). Each ConvBlock includes a BatchNorm2d layer followed by ReLU activation.

Hidden Layers	Output Size	Operations	# of Filters
ConvBlock	13×39	3x3 convolution, padding=1, stride=1	32
Pooling	6×19	2x2 max pooling, stride=2	-
ConvBlock	6×19	3x3 convolution, padding=1, stride=1	64
Pooling	3×9	2x2 max pooling, stride=2	-
Dropout	-	Dropout with $p = 0.25$	-
Flatten	1278	-	-
Fully Connected layer	128	ReLU activation	128
Dropout	-	Dropout with $p = 0.5$	-
Fully Connected layer	10	-	10

4 and 5 respectively. After obtaining the encodings from both the Image encoder and Audio encoder, we concatenated the output and passed it toward fully connected layers (classifier) to generate the final prediction, where the detail can be found 6.

2.3 Model Training

We use cross-entropy loss as our loss function to optimize throughout all the models. For each model, we set for different hyperparameters such as the learning rate and the number of epochs to be trained for. We use Adam as the optimizer with a weight decay of 1e-5. We also use ReduceLROnPlateau to reduce the learning rate (patience=5) if the validation accuracy is not decreasing. An early stopping mechanism is employed with the patience of 10. Here are the training hyperparameters for all the models:

- ImageNet_CNN: lr=1e-3, epochs=10, batch size=32
- AudioNet_CNN: lr=1e-4, epochs=100, batch size=64
- AudioNet_LSTM: lr=5e-4, epochs=50, batch size=64
- HybridNet_CNN: lr=1e-3, epochs=20, batch size=32
- HybridNet_CNN_LSTM: lr=1e-4, epochs=30, batch size=64

Table 3: Audio-Based Model (AudioNet_LSTM). The model utilizes LSTM layers followed by fully connected layers for classification. The input is 39x13 where 39 is the sequence length and 13 is the feature size.

Hidden Layers	Output Size	Operations	# of Units/Filters
LSTM	64	2 layers	64
Fully Connected layer	64	ReLU activation	64
Fully Connected layer	10		10

Table 4: The encoder part for the Multimodal Model (HybridNet_CNN). Each ConvBlock includes a BatchNorm2d layer followed by ReLU activation.

	Hidden Layers	Output Size	Operations	# of Filters
Image Encoder	ConvBlock	28×28	3x3 convolution, padding=1, stride=1	32
	Pooling	14×14	2x2 max pooling, stride=2	-
	ConvBlock	14×14	3x3 convolution, padding=1, stride=1	64
	Pooling	7×7	2x2 max pooling, stride=2	-
	Dropout	-	Dropout with $p = 0.25$	-
	Flatten	3136	-	-
	Fully Connected layer	128	ReLU activation	128
	Dropout	-	Dropout with $p = 0.5$	-
Audio Encoder	ConvBlock	13×39	3x3 convolution, padding=1, stride=1	32
	Pooling	6×19	2x2 max pooling, stride=2	-
	ConvBlock	6×19	3x3 convolution, padding=1, stride=1	64
	Pooling	3×9	2x2 max pooling, stride=2	-
	Dropout	-	Dropout with $p = 0.25$	-
	Flatten	1278	-	-
	Fully Connected layer	128	ReLU activation	128
	Dropout	-	Dropout with $p = 0.5$	-

Table 5: The encoder part for the Multimodal Model (HybridNet_CNN_LSTM). Each ConvBlock includes a BatchNorm2d layer followed by ReLU activation.

	Hidden Layers	Output Size	Operations	# of Filters
Image Encoder	ConvBlock	28×28	3x3 convolution, padding=1, stride=1	32
	Pooling	14×14	2x2 max pooling, stride=2	-
	ConvBlock	14×14	3x3 convolution, padding=1, stride=1	64
	Pooling	7×7	2x2 max pooling, stride=2	-
	Dropout	-	Dropout with $p = 0.25$	-
	Flatten	3136	-	-
	Fully Connected layer	128	ReLU activation	128
	Dropout	-	Dropout with $p = 0.5$	-
Audio Encoder	LSTM	64	2 layers	64
	Fully Connected layer	128	ReLU activation	128
	Dropout	-	Dropout with $p = 0.25$	-

Table 6: The classifier part for the Multimodal Model. Both models share the same classifier design. The input shape is 256 dimensional as it is the concatenation of the image embeddings and audio embeddings.

Hidden Layers	Output Size	Operations	# of Filters
Fully Connected layer	128	BatchNorm1D + ReLU activation	128
Dropout	-	Dropout with $p = 0.25$	-
Fully Connected layer	10	-	10

2.4 Hyperparameter Tuning

We performed a grid search between the number of parameters that each model has, such as the number of layers of CNN layers (2 or 3 layers) or the number of filters (32,64,128...). For the learning rate and epochs, we tried ranging between $(1e^{-3}, 1e^{-5})$ and $(10, 100)$ respectively. The configuration with the best accuracy is then chosen for the model.

3 Results

Here we show the results of test data for all the models. We use the accuracy and f1 score with macro averaging to evaluate the result. Since we split the training data into a ratio of 14:3:3. The size of the test data is 9000. From Table 7, we can see that the accuracy and f1 score for both the multimodal model models (HybridNet_CNN and HybridNet_CNN_LSTM) outperformed those that used only one kind of input data by a margin. However, we can also see that the LSTM-based model performs slightly worse when used to encode the audio data. One possible reason is that LSTMs

Table 7: The accuracy and f1 score of the test set for all the models.

Models	Accuracy	f1 score
ImageNet_CNN	99.13%	0.991
AudioNet_CNN	97.75%	0.978
AudioNet_LSTM	97.18%	0.972
HybridNet_CNN	**99.74%**	**0.997**
HybridNet_CNN_LSTM	99.36%	0.994

are designed to capture long-term dependencies in sequence data, which might not be as critical or well-represented in the structure of MFCCs as spatial relationships which CNNs excel at capturing.

Figure 4 displays a set of ambiguous image data. It is evident that AudioNet_CNN successfully classifies these samples by leveraging audio data, which proves more distinguishable in these scenarios. The multimodal approach, HybridNet_CNN, also achieves correct predictions. However, it can still be influenced by the image data; for instance, as seen in column 2, the output prediction mirrors that of the ImageNet_CNN model.

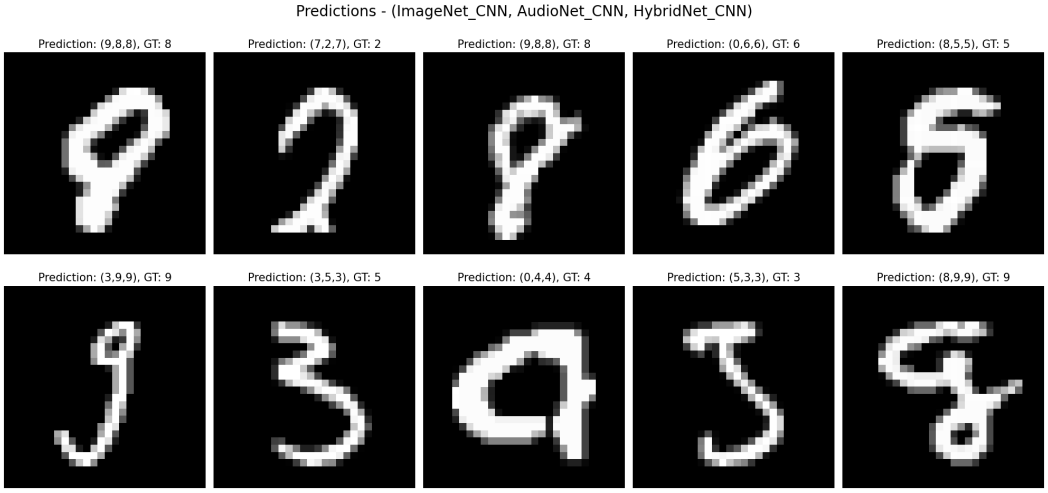


Figure 4: Here are examples where the ImageNet_CNN model predicted incorrectly. The sequence of models shown, from left to right, is ImageNet_CNN, AudioNet_CNN, and HybridNet_CNN. It is evident that in these instances, incorporating audio features leads to correct predictions by both the Audio model and the Hybrid model.

To further illustrate these results, we refer to the Venn diagram in Figure 5, which displays the indices of samples each model predicts incorrectly. Notably, there is no overlap between AudioNet_CNN and HybridNet_CNN, indicating that when pure audio input leads to incorrect predictions, the integration of image data in the multimodal model helps to clarify these ambiguities. A similar effect is observed with image-only data. Originally, ImageNet_CNN incorrectly predicted 63 samples, but with the addition of audio data, this number decreases to 15, affecting only 8 samples originally predicted correctly by ImageNet_CNN.

Finally, we perform dimension reduction methods to visualize the embedding of the training data in a 2-dimensional space. We first perform Principal component analysis (PCA) to reduce the embedding dimensionality to 50, then use tSNE to further reduce it to 2 dimensions. We performed k-means clustering with k=10 to find the 10 clusters. This allows us to observe the clusters formed by different classes and assess the quality of the embeddings in terms of class separability and overlap. Here we show the results for ImageNet_CNN (Figure 6), AudioNet_CNN (Figure 7), and HybridNet_CNN model (Figure 8). We can observe that the embeddings from the ImageNet_CNN, AudioNet_CNN, and HybridNet_CNN's image encoder are very separable after performing dimension reduction into 2-dimensional space. However, for the HybridNet_CNN's audio embedding, although clusters are observable, they are not as distinct when reduced to only two-dimensional space. This suggests that

the audio embeddings may require either higher-dimensional space to maintain separability or a different dimension reduction technique that preserves more of the inherent data structure.

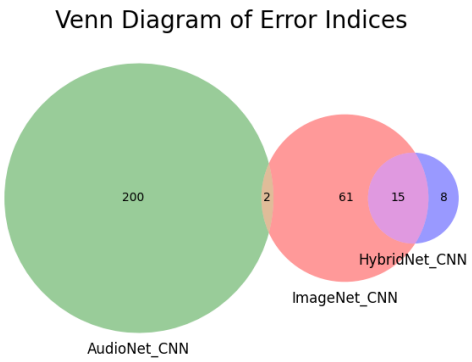


Figure 5: Venn diagram of the error indices on the test data.

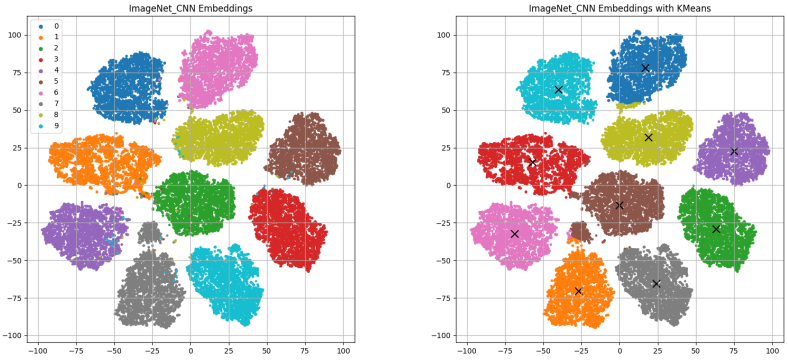


Figure 6: Dimension Reduction result for the ImageNet_CNN model: The left image displays a scatter plot with original labels, while the right image shows the k-means clustering result with k=10.

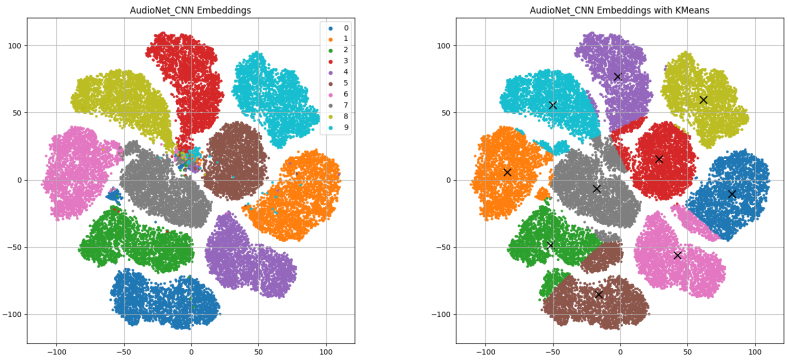


Figure 7: Dimension Reduction result for the AudioNet_CNN model: The left image displays a scatter plot with original labels, while the right image shows the k-means clustering result with k=10.

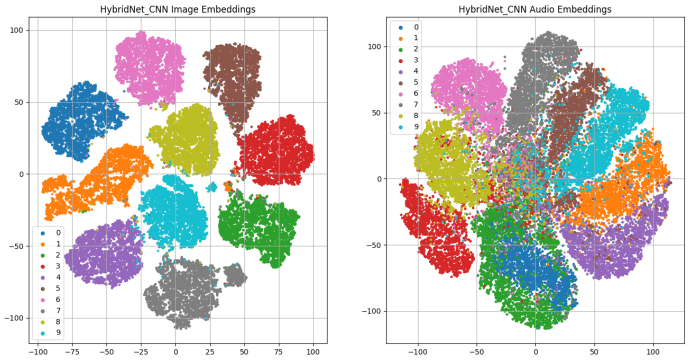
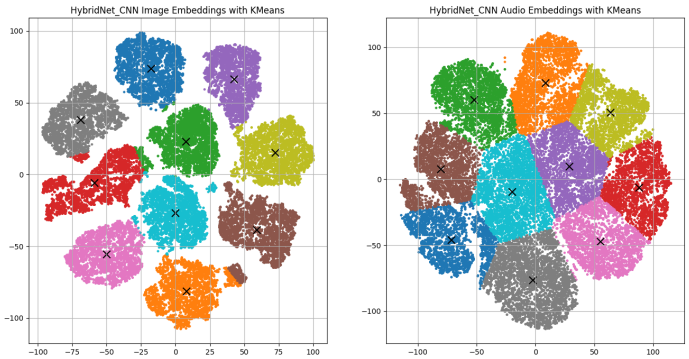


Figure 8: Dimension Reduction result for the HybridNet_CNN model: The top image displays a scatter plot with original labels, while the bottom image shows the k-means clustering result with k=10.



4 Conclusion

In conclusion, this paper demonstrated how the use of a multimodal model enhances classification accuracy on the Written and Spoken Digits Database compared to using solely image or audio data. By leveraging the advantages of diverse input data, the model is less prone to ambiguity. Given these promising results, further research could explore extending these techniques to more complex datasets and scenarios, potentially applying similar multimodal approaches in other domains where data ambiguity can significantly impact performance.

References

[1] Khacef, L., Rodriguez, L., Miramond, B.: Written and spoken digits database for multimodal learning (Oct 2019). https://doi.org/10.5281/zenodo.3515935, `https://doi.org/10.5281/zenodo.3515935`