ASC
CLASSIFICATION

ACOUSTIC SCENE CLASSIFICATION

USING DEEP LEARNING

ABSTRACT

This paper addresses the Acoustic Scene Classification (ASC) task with the aim of surpassing the baseline model accuracy set by the DCASE 2021 Challenge Task 1 - Subtask A, which stands at 47.7%. Leveraging the TAU Urban Acoustic Scenes 2020 Mobile dataset, the study utilizes Dynamic Time Warping (DTW) to condense the dataset to 9,225 audio files while retaining vital information. Two models, a Convolutional Neural Network (CNN) and a Recurrent Neural Network (RNN), are developed and trained using Mel-Frequency Cepstral Coefficients (MFCC) features.

The paper underscores the significance of a structured JSON file, named "data," encapsulating essential dataset details, which not only enhances efficiency but also simplifies the training of both CNN and RNN models. The CNN achieves approximately 54.14% accuracy, while the RNN excels with an accuracy of around 55.12%, demonstrating a notable improvement over the baseline model and affirming the effectiveness of the proposed models in ASC.

Introduction To ASC (Acoustic Scene Classification)

- It is a specialized task focused on classifying audio signals from diverse environmental soundscapes into predefined categories.
- These categories can represent various settings like airports, streets, etc
- ASC encounters inherent complexities in recognizing and categorizing sounds within different environmental contexts. These challenges include:

    1. Distinctive characteristics of audio signals from various scenes,
    2. Interpretation of social and environmental cues embedded in audio data,
    3. Varied patterns in soundscapes, including repetitive elements or unique features

- ASC demands resilient models to accurately categorize audio signals in diverse environments, ensuring flexibility without drawing parallels to other domains.

Literacy Survey

01

I. Martín-Morato et al.

Developed low-complexity ASC models for multi-device audio, achieving an overall accuracy of 47.7% with a log loss of 1.473 on the development dataset.

02

K. Horvath et al.

Utilized ARCFACE Metric Learning to achieve 54-55% accuracy with a log loss of 1.597 on the DCASE 2021 development dataset for ASC.

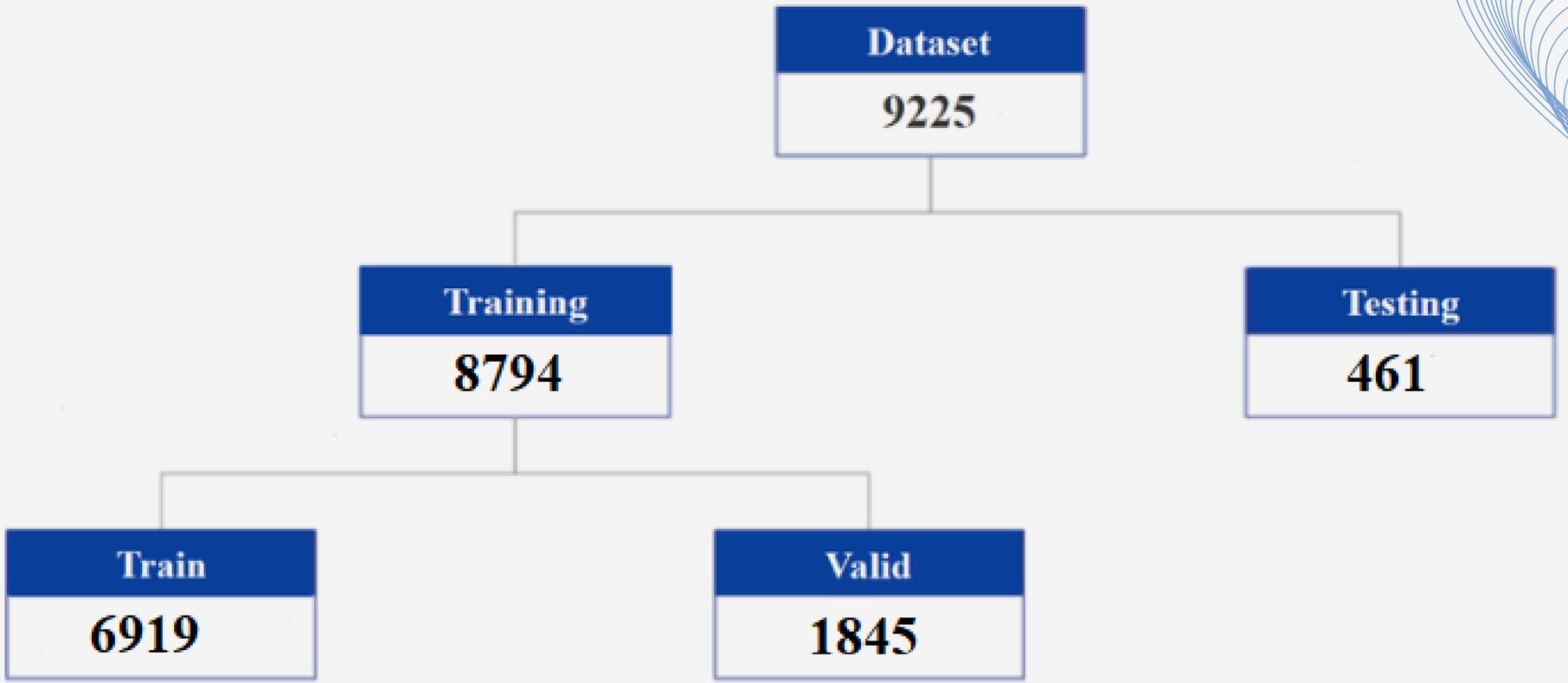
03

A. Singh et al.

Employed pruning and quantization for low-complexity ASC, achieving 47-49% accuracy with log loss ranging from 1.383 to 1.425 on the DCASE 2021 development dataset.

S.N.	Author	Year	Title of Paper	Findings	Relevance
1	I. Martin-Morato et al.	2021	Low-complexity acoustic scene classification for multi-device audio: Analysis of DCASE 2021 Challenge systems	The paper analyzes low-complexity acoustic scene classification for multi-device audio, particularly focusing on systems involved in the DCASE 2021 Challenge.	The findings align with the cornerstones of my project, particularly in understanding challenges and solutions for multi-device audio scenarios, as evidenced by the DCASE 2021 Challenge analysis.
2	K. Horvath et al.	2021	Using ARCFACE Metric Learning For Low-Complexity Acoustic Scene Classification	The paper explores the use of ARCFACE metric learning for low-complexity acoustic scene classification.	The focus on pruning and quantization for low-complexity acoustic scene classification aligns closely with my goal of improving model efficiency. The optimization techniques discussed are valuable for addressing challenges in my research.
3	A. Singh et al.	2021	Pruning and Quantization For Low-Complexity Acoustic Scene Classification	The paper investigates the application of pruning and quantization techniques for achieving low-complexity acoustic scene classification.	The exploration of metric learning techniques provides valuable insights that could complement or enhance my classification methods.

Splitting Dataset



TRAINNING PROCESS

Json File Creation Parameters:

- **Sampling Rate:** 44,100 Hz
- **Duration:** 10 milliseconds
- **No. of MFCC coefficients:** 13
- **No. of segments to divide each audio file:** 5
- **Hop length for consecutive frames of STFT:** 25 milliseconds
- **No. of data points in each Short-Time Fourier Transform:** 2048

The aspects in the training process of model are as follows:

- **Optimizer:** Adam
- **Loss Function:** sparse_categorical_crossentropy
- **Number of Epochs:** 50
- **Batch Size:** 32

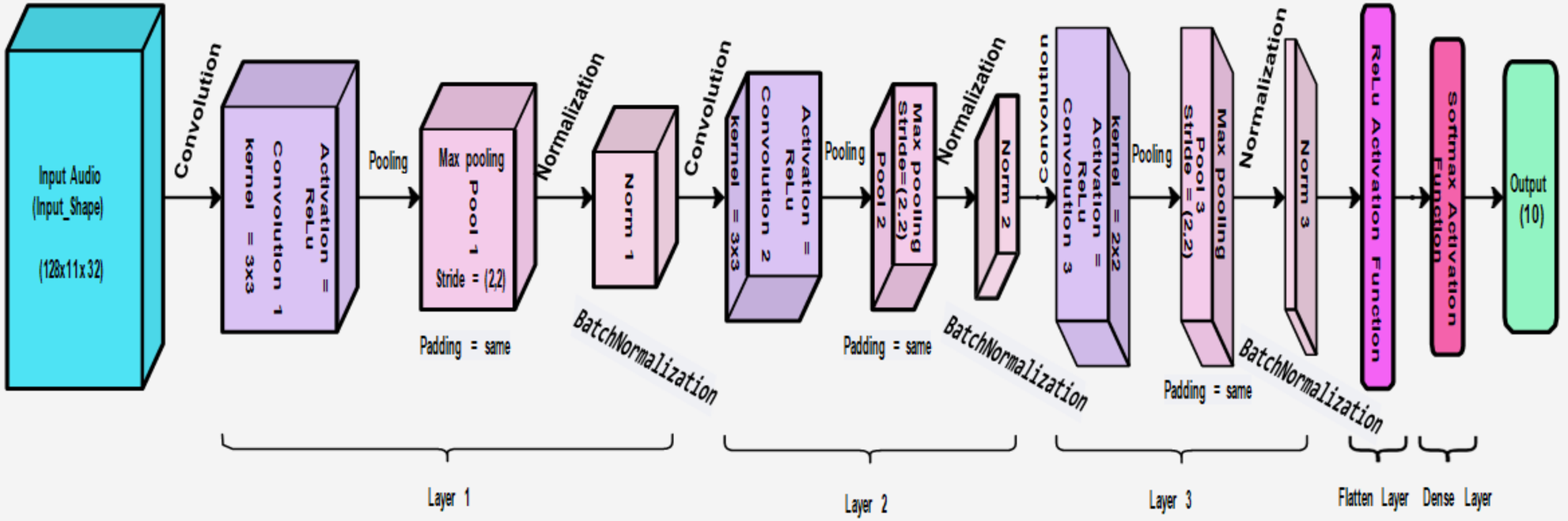
JSON FILE STORED


```
"mapping": [
    "Dataset\\original_audio\\airport",
    "Dataset\\original_audio\\bus",
    "Dataset\\original_audio\\metro",
    "Dataset\\original_audio\\metro_station",
    "Dataset\\original_audio\\park",
    "Dataset\\original_audio\\public_square",
    "Dataset\\original_audio\\shopping_mall",
    "Dataset\\original_audio\\street_pedestrian",
    "Dataset\\original_audio\\street_traffic",
    "Dataset\\original_audio\\tram"
  ],

"mfcc": [
        -263.89422607421875, 123.72734069824219, -10.349501609802246, 30.706527709960938,
        -3.2898082733154297, 8.46491527557373, 21.342144012451172, 5.234488487243652,
        -0.1594223976135254, 10.757991790771484, -2.6126549243927, 10.814064025878906,
        -1.3333773612976074
]

"label": [
        0, 0, 0,....., 1, 1, 1,....., 2, 2, 2,....., 3, 3, 3,....., 4, 4, 4,.....,
        5, 5, 5,....., 6, 6, 6,....., 7, 7, 7,....., 8, 8, 8,....., 9, 9, 9,.....
]
```

CNN Architecture



- This CNN architecture is designed for acoustic scene classification, employing convolutional layers for feature extraction, pooling for spatial reduction, and a fully connected layer for classification.
- Dropout and batch normalization are applied to enhance training stability and prevent overfitting.
- The final output layer uses softmax activation for accurate class predictions.

01

Convolutional Layers

- Three convolutional layers with distinct configurations for feature extraction.
- First layer: 32 filters, 3x3 kernel, ReLU activation, "same" padding.
- Second layer: 32 filters, 3x3 kernel, ReLU activation, "same" padding.
- Third layer: 32 filters, 2x2 kernels for broader feature extraction.

02

Pooling and Normalization

- Max-pooling used for spatial reduction and information compression.
- Batch normalization enhances overall performance during training.

03

Fully Connected Layer

- Flattening layer converts 2D feature maps into 1D vectors.
- Fully connected layer with 64 neurons and ReLU activation.

04

Output Layer

- Output layer with 10 neurons for multi-class classification.
- Softmax activation used for probability distributions over classes.

05

Dropout for Overfitting

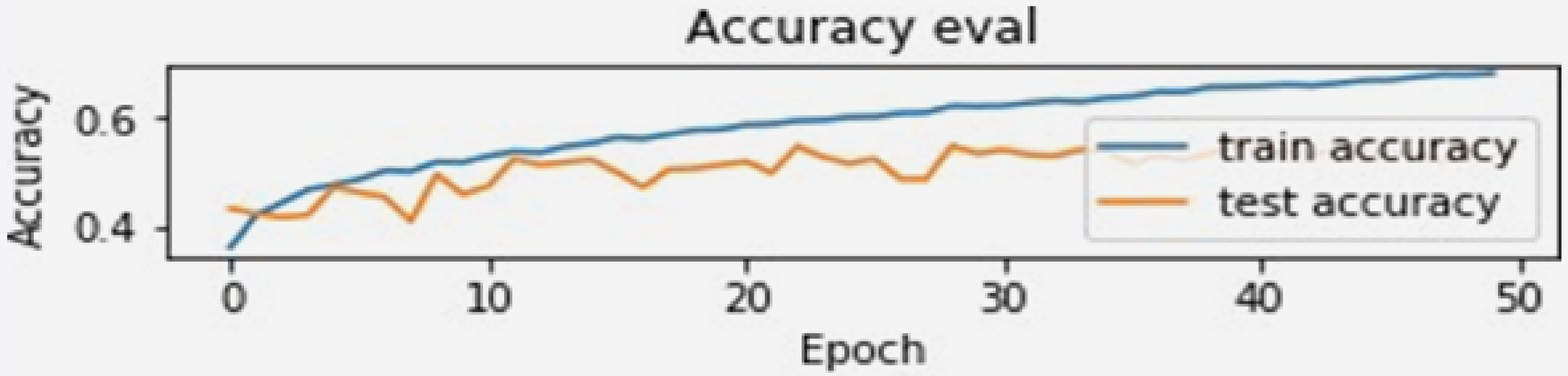
- Dropout is applied during training to mitigate overfitting.
- The dropout rate is set to 30% to randomly deactivate neurons, promoting generalization.
- It helps the model generalize better to unseen data by preventing over-reliance on particular features.

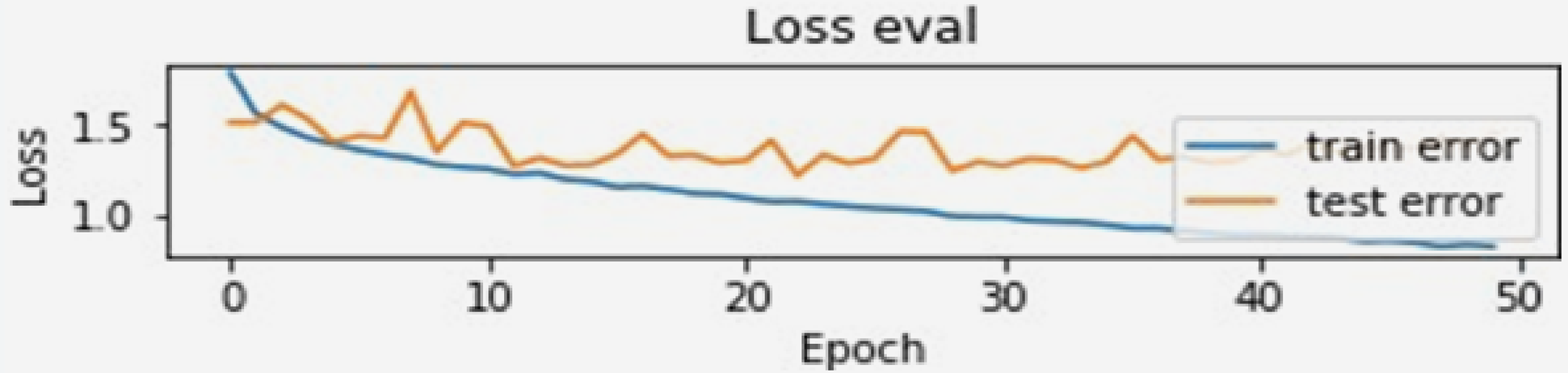
TABLE II: CNN Model Summary

Layer (type)	Output Shape	Param #
conv2d (Conv2D)	(128, 11, 32)	320
max_pooling2d (MaxPooling2D)	(64, 6, 32)	0
batch_normalization (BatchNormalization)	(64, 6, 32)	128
conv2d_1 (Conv2D)	(62, 4, 32)	9248
max_pooling2d_1 (MaxPooling2D)	(31, 2, 32)	0
batch_normalization_1 (BatchNormalization)	(31, 2, 32)	128
conv2d_2 (Conv2D)	(30, 1, 32)	4128
max_pooling2d_2 (MaxPooling2D)	(15, 1, 32)	0
batch_normalization_2 (BatchNormalization)	(15, 1, 32)	128
flatten (Flatten)	(480)	0
dense (Dense)	(64)	30784

Total params : 45,514
Trainable params : 45,322
Non-trainable params : 192

CNN Model Evaluation





Epoch 50/50
1296/1296 [==========] - 112s 86ms/step - loss: 0.9738 - Accuracy: 0.6288 - val_loss: 0.1391 - val_accuracy: 0.5414

RNN Architecture

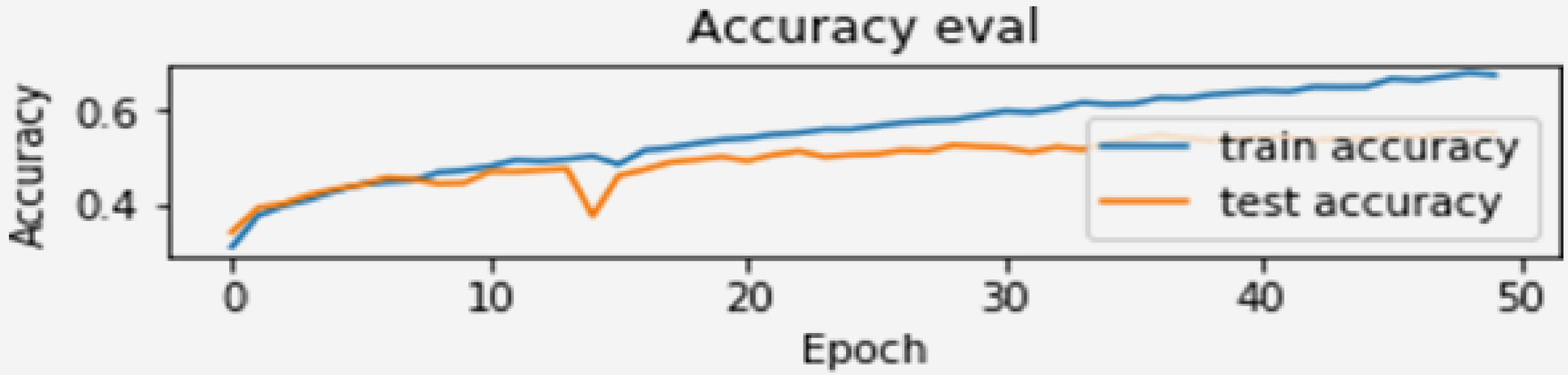
- The RNN model is structured with recurrent layers, enabling effective processing of sequential data and capturing essential temporal patterns in audio.
- Dropout and batch normalization techniques are implemented to ensure training stability, prevent overfitting, and enhance the generalization capabilities of the RNN model.
- The final layer of the RNN employs softmax activation, facilitating precise class predictions and aligning with the unique characteristics of audio sequences.

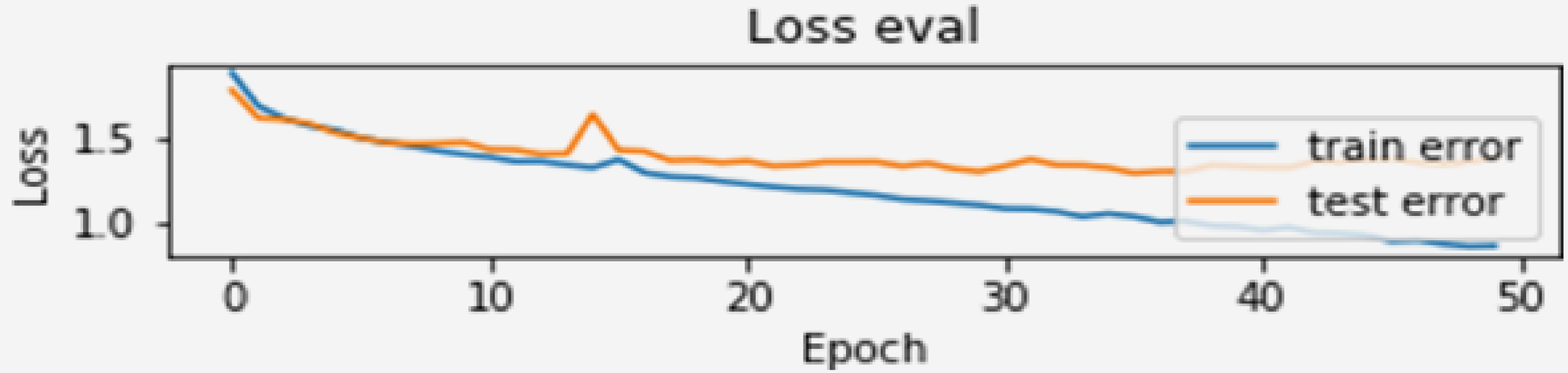
TABLE III: RNN Model Summary

Layer (type)	Output Shape	Param #
lstm (LSTM)	(130, 64)	19968
lstm_1 (LSTM)	(64)	33024
dense_8 (Dense)	(64)	4160
dropout_4 (Dropout)	(64)	0
dense_9 (Dense)	(10)	650

Total params : 57,802
Trainable params:57,802
Non-trainable params:0

RNN Model Evaluation





Epoch 50/50
1296/1296 [==========] - 112s 86ms/step - loss: 0.8438 - Accuracy: 0.5277 - val_loss: 0.1352 - val_accuracy: 0.5512

Result & Analysis

Validation & Testing

Test Dataset

- Another distinct dataset, known as the test dataset, was reserved for final evaluation.
- The test dataset was not used during model training or parameter tuning to ensure an unbiased assessment of the model's performance.
- After the model was trained and validated, it was evaluated on the test dataset to measure its generalization ability and provide an estimate of its performance on unseen data.

Validation Dataset

- A portion of the available data was set aside as the validation dataset.
- During the training process, after each epoch, the model's performance was evaluated on this dataset.
- The validation dataset served as an independent set to monitor the model's progress, detect overfitting, and tune hyperparameters if necessary.

Validation & Testing

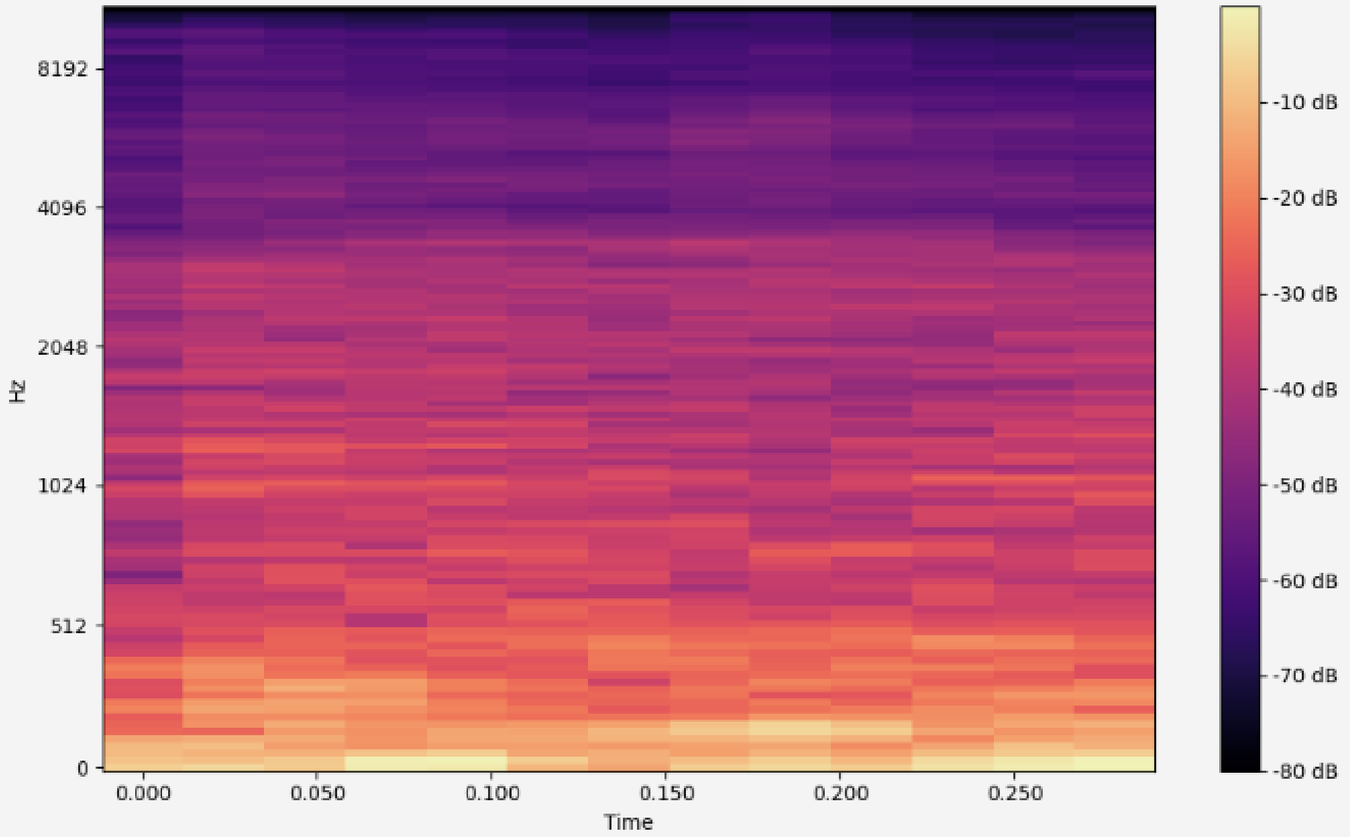
Model Name	Miss Classification Rate		
	Training	Validation	Test
CNN	0.0140	0.0625	0.0342
RNN	0.00112	0.0644	0.0376

- By utilizing separate validation and test datasets, the model's ability to generalize and perform well on new, unseen data was assessed.

- This approach helps ensure that the model can accurately classify or predict outcomes beyond the data it was trained on, providing insights into its real-world performance.

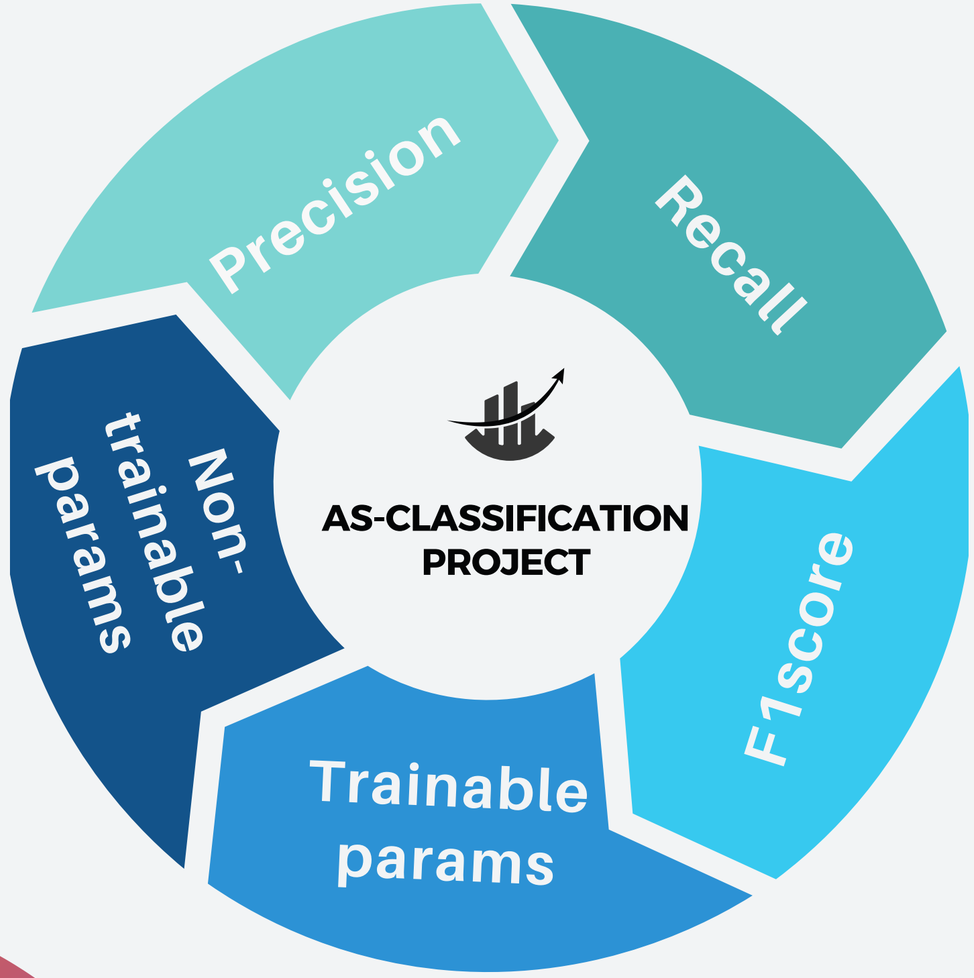
PREDICTION

```
1/1 [==============================] - 0s 43ms/step
+-----------+--------+--------------+
| Class     | City   | Percentage   |
+===========+========+==============+
| Barcelona | Lisbon | 100.00%      |
+-----------+--------+--------------+
Overall Probability: 1.0
```



Predicted Probabilities: 1.0

PRECISION, RECALL, F1SCORE



- The precision, recall, and F1 score values are performance metrics calculated from the model's predictions.
- These metrics are typically calculated based on the comparison between the predicted labels and the true labels of the samples in the dataset.

Model Name	Precision	Recall	F1 Score
CNN	0.5634	0.5450	0.5394
RNN	0.5778	0.5743	0.5579

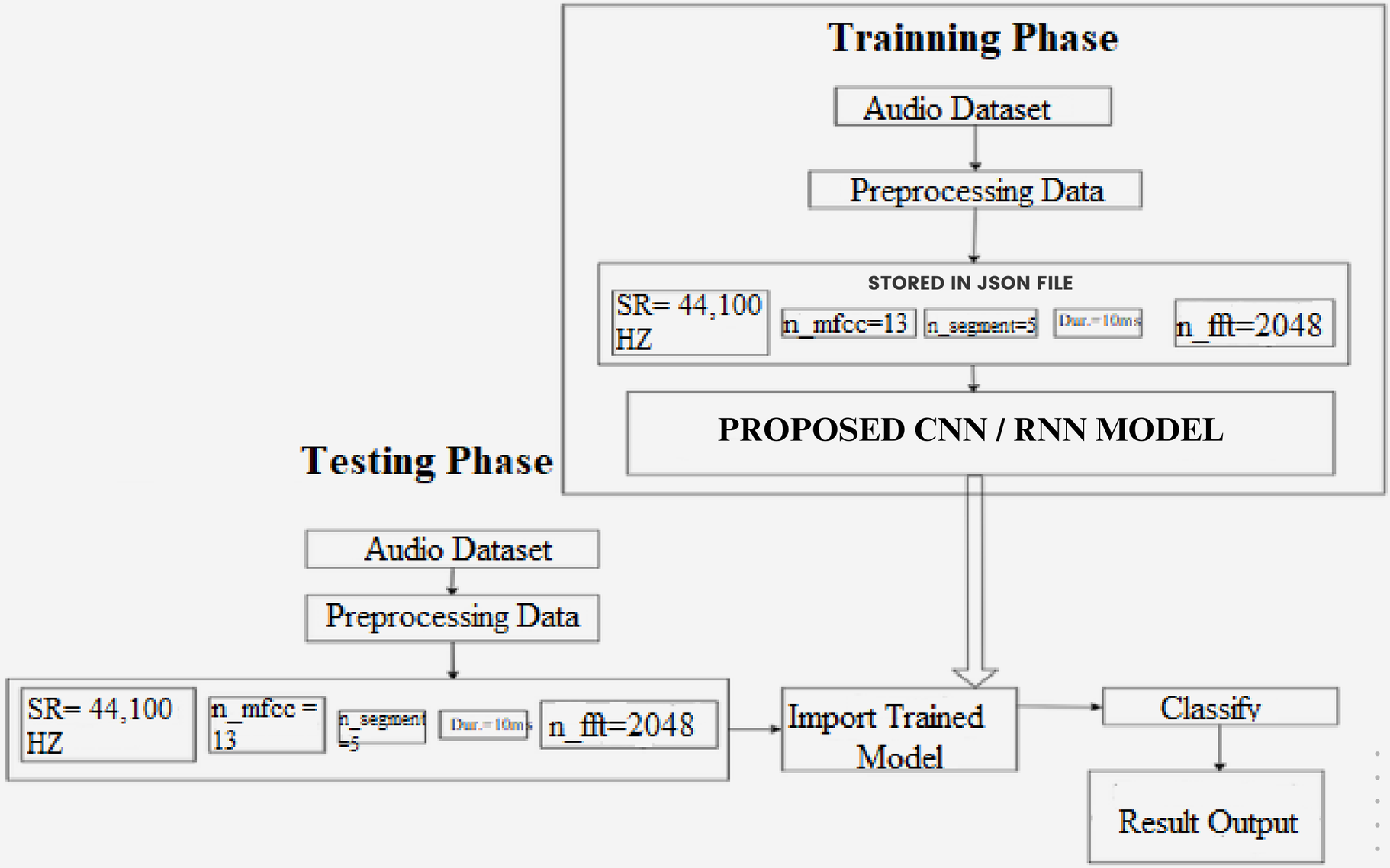
COMPARISON

- Although some deep learning algorithms have been proposed to detect and classify ASC, the datasets they have used are all from IEEE.

Literature	Validation Accuracy	Log Loss
I. Martin-Morato et al.	0.477	1.730
A. Singh et al.	0.489	1.383
Proposed CNN	0.542	1.391
Proposed RNN	0.551	1.352

- When we compared our model with another established model of deep RNN & CNN architecture with TAU Urban Acoustic Scenes 2020 Mobile Development Dataset, our model achieved a higher level of accuracy in both Validation and Log Loss.

Flowchart of Acoustic Scene Classification

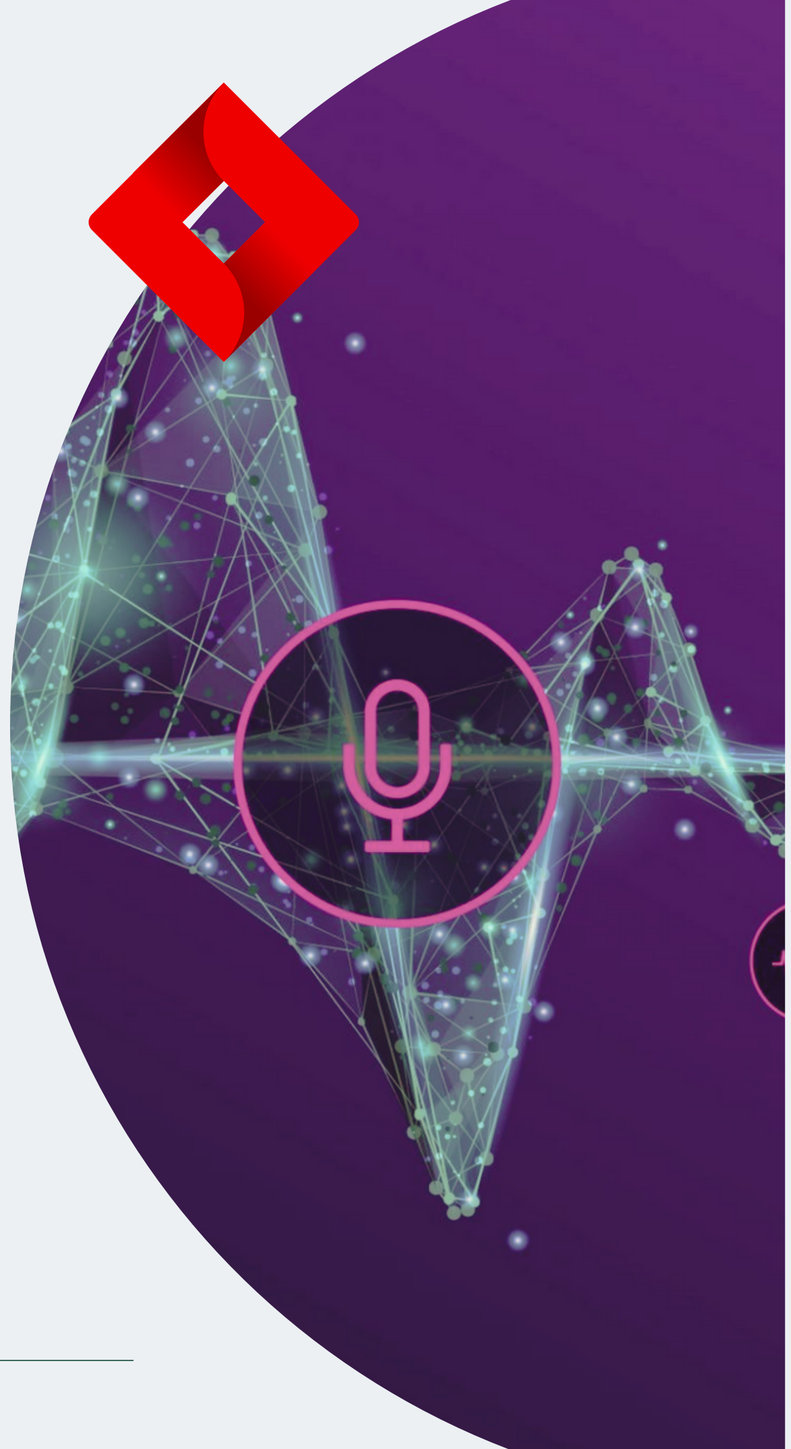


FUTURE PLANS

- Experiment with various CNN architectures to enhance audio classification accuracy.
- Fine-tune parameters like learning rate, batch size, and regularization for improved model performance.
- Implement techniques to artificially diversify the training dataset, boosting the model's ability to generalize.
- Explore transfer learning by using pre-trained models on larger audio datasets, fine-tuning for specific classification tasks.
- Develop a real-time inference system for practical applications, considering deployment on mobile or edge devices.

CONCLUSION

In this groundbreaking project, we've advanced audio classification precision through the strategic use of CNN & RNN and MFCC features, thoughtfully stored in a JSON file. Our adept data extraction and representation have established the bedrock for a robust model. As we look ahead, our focus will be on fine-tuning architecture, optimizing hyperparameters, exploring transfer learning, and ensuring real-time applicability—all driven by the rich information stored in the MFCC features. This initiative not only propels audio classification capabilities but also unlocks avenues for practical deployments in real-world scenarios, underscoring the seamless integration of innovation and application



References

1. I. Mart´ın-Morato´, T. Heittola, A. Mesaros, and T. Virtanen, ”Low- complexity acousticscene classification for multi-device audio:Analysis of DCASE 2021 Challenge systems,” arXiv preprint arXiv:2105.13734, 2021.

2. A. Singh, D. V. Devalraju, and P. Rajan, ”Pruning and Quantization For Low-Complexity Acoustic Scene Classification,”Detection and Classifi- cation of Acoustic Scenes and Events,2021.

3. V. K. Singh, K. Sharma,and S. N. Sur, ”A survey on preprocessing and classification techniques for acoustic scene,” Expert Systems with Applications, 120520, 2023

4. G. Sharma, K. Umapathy, S.Krishnan, ”Trends inaudio sig- nal feature extraction methods,” Applied Acoustics158 107020. doi:10.1016/j.apacoust.2019.107020, 2020.

5. A. Singh, D. V. Devalraju, and P. Rajan, ”Pruning and Quantization For Low-Complexity Acoustic Scene Classification,”Detection and Classifi- cation of Acoustic Scenes and Events,2021.

6. Z. Mushtaq, and S. F. Su, ”Efficient classification of environmental sounds throughmultiple features aggregation and data enhancement techniques for spectrogram images,”Symmetry, 12(11):1822, 2020.