

Comparative Analysis of MFCC and MFE Feature Extraction Methods for CNN-Based Railway Sound Classification

Nadaval Hagga¹, Sapto Wibowo², Moechammad Sarosa³

^{1,2,3} Department of Electrical Engineering, State Polytechnic of Malang, 65141, Indonesia.

¹nadavalhagga16@gmail.com, ²sapto.wibowo@polinema.ac.id, ³msarosa@polinema.ac.id

Abstract— Environmental sound classification is one of the artificial intelligence fields widely applied in embedded system-based monitoring applications. This study compares two audio feature extraction methods, namely Mel-Frequency Cepstral Coefficients (MFCC) and Mel-Filterbank Energy (MFE), for railway sound classification using a Convolutional Neural Network (CNN) model. The dataset consisted of 153 audio samples collected directly from real-world environments, comprising 80 railway sound samples and 73 environmental noise samples. The dataset was divided into 122 training samples and 31 testing samples. Feature extraction and model training were performed using the Edge Impulse platform, and the trained models were subsequently implemented on a Raspberry Pi 5. The training results showed that MFCC achieved 96.7% accuracy, 97% precision, 97% recall, and 97% F1-score, while MFE achieved 98.4% accuracy, 98% precision, 98% recall, and 98% F1-score. During implementation on the Raspberry Pi 5, the MFCC-based model achieved 96.77% accuracy, while the MFE-based model achieved 100% accuracy. However, the MFCC-based model demonstrated lower computational latency, with an average inference time of 12.317 ms, compared to 23.44 ms for the MFE-based model. These results indicate that MFE provides higher classification performance, whereas MFCC provides better computational efficiency in terms of inference time, demonstrating a trade-off between classification performance and computational efficiency for railway sound classification on embedded platforms.

Keywords— *Railway sound classification, MFCC, MFE, CNN, Raspberry Pi 5*

I. INTRODUCTION

Environmental sound classification has emerged as an important topic in artificial intelligence and digital signal processing due to its ability to automatically identify and categorize acoustic events from surrounding environments. By analyzing sound patterns, this technology supports a wide range of applications, including intelligent surveillance, activity recognition, transportation monitoring, and embedded systems that require real-time acoustic perception [1], [2].

The rapid development of machine learning and deep neural network techniques has substantially enhanced the capability of environmental sound classification systems to recognize complex acoustic patterns. The effectiveness of these systems is strongly influenced by the quality of audio feature representation, since feature extraction determines how relevant information is represented before being processed by the classification model. Because environmental sounds contain discriminative information in both the temporal and spectral domains, an appropriate feature representation is essential for achieving reliable classification performance [3].

Among various audio feature extraction techniques, Mel-Frequency Cepstral Coefficients (MFCC) and Mel-Filterbank Energy (MFE) are two commonly used approaches. MFCC represents audio signals using the perceptual Mel-frequency scale and applies cepstral analysis to generate compact feature representations, making it widely used in speech recognition and environmental sound classification [4]. In contrast, MFE directly represents the energy distribution across Mel-scale

filter banks without applying the Discrete Cosine Transform (DCT), thereby retaining spectral energy information in the extracted features [5]. The differences in these feature representations may affect classification performance and computational requirements when applied to environmental sound classification systems.

Practical deployment of environmental sound classification systems, particularly for railway monitoring, introduces several challenges that are rarely encountered under controlled laboratory conditions. Background noise, variations in the distance between the microphone and the sound source, fluctuations in sound intensity, and continuously changing environmental conditions may alter the characteristics of captured audio signals. These factors can affect classification performance and therefore require an appropriate feature representation for distinguishing railway sounds from surrounding environmental noise [6]. Railway sounds exhibit distinctive acoustic characteristics consisting of low- and high-frequency components generated by the engine, wheels, and wheel-rail interactions. Consequently, an effective audio feature representation is required to capture these acoustic patterns

Based on these challenges, this study presents a comparative evaluation of MFCC and MFE feature extraction methods for railway sound classification using a Convolutional Neural Network (CNN) developed on the Edge Impulse platform. Both feature extraction methods are evaluated using the same CNN architecture and experimental conditions to

investigate their influence on classification performance in terms of accuracy, precision, recall, F1-score, and inference time. The trained models are subsequently deployed on a Raspberry Pi 5 to evaluate their on-device classification performance and computational efficiency. The comparison provides an analysis of the trade-off between classification performance and inference time for railway sound classification on an embedded platform.

II. METHOD

This study aims to compare two audio feature extraction methods, namely Mel-Frequency Cepstral Coefficients (MFCC) and Mel-Filterbank Energy (MFE), for railway sound classification on an embedded system. The Edge Impulse platform was used for feature extraction, model training, and classification performance evaluation. The trained models were subsequently deployed on a Raspberry Pi 5 to perform on-device inference. System performance was evaluated using accuracy, precision, recall, F1-score, and inference time.

A. Research Design

The proposed system was developed to classify railway sounds and environmental noise by comparing two audio feature extraction techniques, namely Mel-Frequency Cepstral Coefficients (MFCC) and Mel-Filterbank Energy (MFE). Both methods were implemented within the same environmental sound classification framework to investigate the influence of different feature representations on the performance of a Convolutional Neural Network (CNN) in recognizing railway sounds and environmental noise [7], [8].

To ensure a fair comparison, both feature extraction methods were evaluated using an identical CNN architecture, training configuration, and testing dataset. Under these controlled experimental conditions, differences in classification performance can be attributed primarily to the characteristics of the extracted features rather than variations in the classification model.

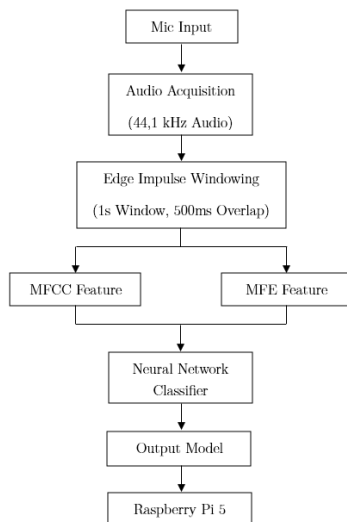


Figure 1. Block diagram system

Figure 1 presents the overall workflow of the proposed railway sound classification system. Audio signals were first captured using a USB microphone with a sampling frequency of 44.1 kHz. The recorded audio was then segmented into 3-second clips before being processed using either the MFCC or MFE feature extraction method. The extracted features were subsequently used as inputs to the CNN classifier for model training and classification. The trained models were then deployed on a Raspberry Pi 5 for on-device railway sound classification.

B. Dataset Audio

The audio dataset was prepared specifically for the development and evaluation of the proposed railway sound classification system. It consists of two categories: railway sounds and environmental noise. Railway sound recordings were collected directly near railway tracks during train operations, whereas environmental noise recordings represent various non-railway acoustic events encountered around the study area. Collecting data under different environmental conditions was intended to represent the acoustic conditions encountered during practical deployment [9], [10].

All recordings were acquired using a USB conference microphone with a sampling frequency of 44.1 kHz. Since the duration of train passages naturally varies, the recorded audio files were segmented into uniform 3-second clips during the preprocessing stage. Railway sound samples were obtained while trains were passing the observation point, whereas environmental noise samples were recorded under diverse ambient conditions to provide representative background acoustic patterns.



Figure 2. Railway Sound Data Collection Process

Figure 2 illustrates the railway sound acquisition process conducted directly beside the railway track. The recording position was selected to capture authentic acoustic signals generated during train operations while simultaneously preserving the influence of surrounding environmental conditions. This direct field-recording approach provides a realistic dataset that closely represents the conditions expected during real-time deployment.

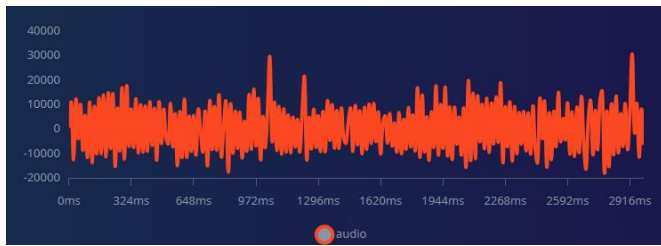


Figure 3. Example of environmental noise waveform

Figure 3 presents an example of the waveform corresponding to an environmental noise recording after segmentation into a 3-second audio clip. Compared with railway sound recordings, the waveform exhibits lower signal amplitude and irregular fluctuations over time, reflecting the random characteristics of surrounding environmental sounds. The vertical axis represents the digital audio amplitude ranging approximately from $-20,000$ to $+40,000$, while the horizontal axis represents the recording duration of approximately 3 seconds (0–2916 ms). These waveform characteristics indicate that environmental noise generally lacks the repetitive high-energy patterns typically observed in railway sound recordings.



Figure 4. Example of railway sound waveform

Figure 4 presents an example of the raw waveform of the railway sound class. Compared with the environmental noise waveform, the railway signal has substantially larger amplitudes, reaching approximately $\pm 35,000$ digital amplitude units throughout most of the recording duration. The waveform also shows a more continuous and periodic oscillatory pattern, indicating stronger acoustic energy generated by the train. These characteristics provide richer spectral information that can later be extracted using MFCC and MFE feature extraction methods.

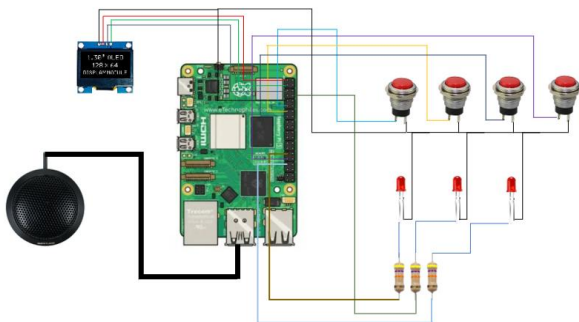


Figure 5. Hardware implementation of the railway sound classification

Figure 5 shows the hardware configuration of the proposed railway sound classification system based on Raspberry Pi 5. The system consists of a USB microphone for audio acquisition, a 1.3-inch OLED display for displaying the classification results, four push buttons for system control, and three LEDs as status indicators. The OLED communicates with the Raspberry Pi 5 through the I²C interface using the SDA and SCL pins, while the microphone is connected through the USB interface. The push buttons are used to control data recording, system shutdown, and real-time detection, whereas the LEDs provide visual feedback regarding the current operating state of the system.

GPIO	Function
17	Railway sound recording button
27	Environmental noise recording button
22	Shutdown button
6	Start detection button
5	Railway recording LED
14	Environmental noise recording LED
13	Power LED
SDA	OLED SDA
SCL	OLED SCL
VCC	OLED power supply
GND	Common ground

The GPIO configuration used in this implementation is summarized as GPIO17 is assigned to record railway sound samples, GPIO27 is used to record environmental noise samples, GPIO22 functions as the system shutdown button, and GPIO6 is used to initiate the real-time detection process. Three LEDs are connected to GPIO5, GPIO14, and GPIO13 to indicate railway recording status, environmental noise recording status, and system power status, respectively. The OLED display is connected through the I²C interface using the SDA, SCL, VCC, and GND pins.

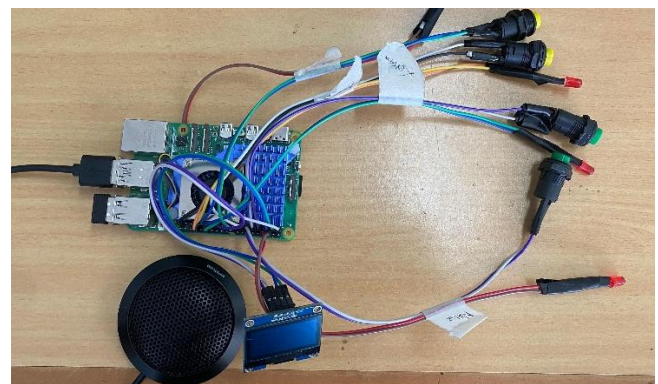


Figure 6. Raspberry Pi 5 Used in This Study

Figure 6 shows the hardware configuration of the proposed railway sound classification system. The system consists of a Raspberry Pi 5 as the embedded computing platform, a USB

microphone for audio acquisition, a 1.3-inch OLED display for presenting classification results, four push buttons for system control, and three LEDs used as status indicators. The Raspberry Pi 5 performs both audio data acquisition and on-device inference using the trained CNN models exported from the Edge Impulse platform. The OLED display provides real-time visualization of the predicted probabilities for the railway and environmental noise classes, while the buttons are used to record railway and environmental noise samples, initiate the detection process, and safely shut down the system. The LEDs provide visual feedback indicating the recording status and power condition of the system

The recorded audio files were stored in WAV format and subsequently segmented into 3-second audio clips. This segmentation process was performed to obtain uniform-length audio samples for model training and testing using the Edge Impulse platform. After data preparation, a total of 153 audio samples were obtained, consisting of 80 railway sound samples and 73 environmental noise samples. The dataset was divided into training and testing subsets. The training set consisted of 122 samples, including 65 railway sound samples and 57 environmental noise samples, while the testing set consisted of 31 samples, including 15 railway sound samples and 16 environmental noise samples. The testing data were kept separate from the training data to evaluate the model using previously unseen audio samples.

C. Audio Feature Extraction

In sound classification systems, feature extraction is responsible for converting raw acoustic signals into discriminative numerical features that serve as inputs to machine learning models. This process aims to preserve the essential characteristics of the audio signal while reducing data complexity, thereby enabling more effective classification [11]. In this study, two audio feature extraction methods, namely Mel-Frequency Cepstral Coefficients (MFCC) and Mel-Filterbank Energy (MFE), were evaluated and compared.

1) Mel-Frequency Cepstral Coefficients (MFCC)

Mel-Frequency Cepstral Coefficients (MFCC) have been extensively adopted in speech recognition and environmental sound classification because they effectively capture the spectral properties of audio signals. By employing the perceptual Mel-frequency scale, MFCC provides a feature representation that closely resembles the frequency sensitivity of the human auditory system. By employing the Mel scale, lower-frequency components are represented with higher resolution than higher-frequency components, enabling more effective representation of the acoustic characteristics of audio signals [12], [13], [14].

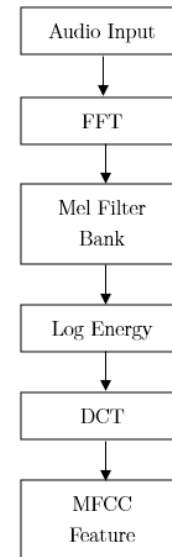


Figure 7. MFCC Feature Extraction Process

Figure 7 illustrates the MFCC feature extraction process. First, The MFCC extraction process begins by converting the input audio signal from the time domain into its frequency-domain representation through the Fast Fourier Transform (FFT). The frequency spectrum is subsequently passed through a bank of Mel-scale filters that emulate the frequency resolution of human hearing. The energy produced by each Mel filter is then converted to a logarithmic scale, after which the Discrete Cosine Transform (DCT) is applied to obtain a compact set of cepstral coefficients used as the final feature representation for audio classification.

TABLE II
MFCC PARAMETER CONFIGURATION IN EDGE IMPULSE

Parameter	Value
Number of Coefficients	13
Frame Length	0.025
Frame Stride	0.02
Filter Number	32
FFT Length	2048
Normalization Window Size	151
Low Frequency	80
High Frequency	Not Set
Pre-emphasis Coefficient	0.98

In this study, MFCC feature extraction was performed using the MFCC processing block available in the Edge Impulse. The configuration parameters included 13 coefficients, a frame length of 0.025 s, a frame stride of 0.02 s, 32 Mel filters, an FFT length of 2048, a lower frequency limit of 80 Hz, and a pre-emphasis coefficient of 0.98. The complete MFCC parameter configuration is presented in Table II.

The output of the MFCC extraction stage is a matrix composed of cepstral coefficients that characterizes the time-varying spectral content of the input audio signal. Each row denotes a specific MFCC coefficient, whereas the columns correspond to consecutive time frames within the analyzed

audio segment. A visualization of the extracted MFCC features is presented in Figure 8.

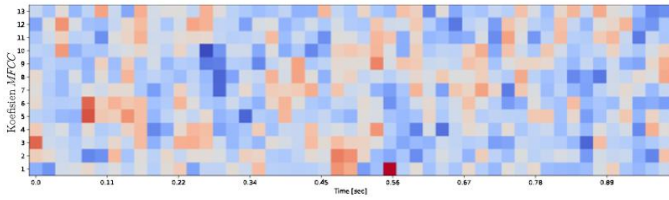


Figure 8. MFCC Feature Extraction Results

Figure 8 displays the MFCC feature representation as a heatmap, where the extracted cepstral coefficients are visualized across successive time frames. The horizontal axis denotes the temporal evolution of the analyzed audio signal, whereas the vertical axis corresponds to the 13 MFCC coefficients. Color variations indicate the magnitude of each coefficient, with warmer colors (red) representing higher values and cooler colors (blue) indicating lower values. The variation in color patterns across the coefficients reflects changes in the spectral characteristics of the audio signal throughout its duration. These coefficient patterns serve as input features for the neural network model to distinguish between railway sounds and environmental noise.

2) Mel-Filterbank Energy (MFE)

Mel-Filterbank Energy (MFE) is an audio feature extraction method that utilizes the energy of each Mel filter to represent the characteristics of audio signals. Unlike MFCC, which applies the Discrete Cosine Transform (DCT) to generate cepstral coefficients, MFE directly preserves the energy values produced by the Mel filter bank, thereby retaining the frequency-domain energy information [15], [16]. The direct representation of Mel-filterbank energy results in a different feature dimensionality from MFCC, which may affect the computational requirements of the classification model, making it well suited for audio classification systems deployed on embedded and edge computing platforms.

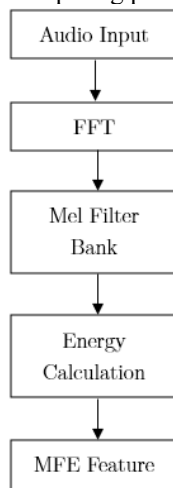


Figure 9. MFE Feature Extraction Process

Figure 9 summarizes the extraction procedure of the Mel-Filterbank Energy (MFE) features. Initially, the input audio waveform is converted into its frequency-domain

representation using the Fast Fourier Transform (FFT). The obtained spectrum is subsequently passed through a bank of Mel-scale filters that partition the frequency components according to human auditory perception. Unlike MFCC, the MFE method directly retains the logarithmic energy of each Mel filter without performing the Discrete Cosine Transform (DCT). Consequently, the extracted feature vector preserves the energy distribution across the Mel filter banks and is supplied directly to the CNN model for railway sound and environmental noise classification.

TABLE III

MFE PARAMETER CONFIGURATION IN EDGE IMPULSE	
Parameter	Value
Frame Length	0.025
Frame Stride	0.01
Filter Number	43
FFT Length	2048
Low Frequency	80
High Frequency	Not Set
Normalization Noise Floor (dB)	-52

In this study, MFE feature extraction was performed using the MFE processing block provided by the Edge Impulse platform. The configuration parameters included a frame length of 0.025 s, a frame stride of 0.01 s, 43 Mel filters, an FFT length of 2048, a lower frequency limit of 80 Hz, and a noise floor of -52 dB. The complete MFE parameter configuration is presented in Table III. The extracted MFE features are represented as the energy distribution across multiple Mel frequency bands over time.

A visualization of the extracted MFE features is presented in Figure 10.

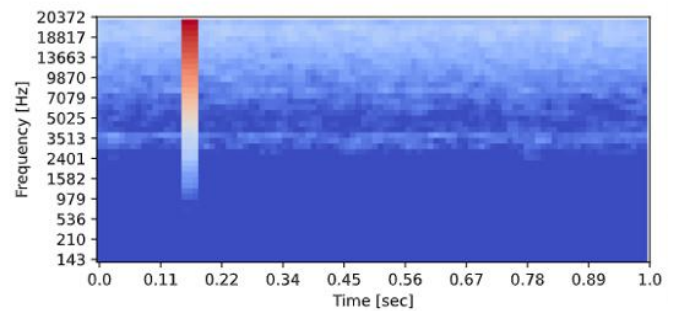


Figure 10. MFE Feature Extraction Results

Figure 10 presents the extracted Mel-Filterbank Energy (MFE) features in the form of a heatmap, illustrating the distribution of audio signal energy across multiple Mel frequency bands over time. The horizontal axis represents the temporal progression of the audio signal, while the vertical axis corresponds to the frequency bands generated by the Mel filter bank. The color intensity indicates the magnitude of the energy within each frequency band, where red represents higher energy levels and blue represents lower energy levels. As shown in the figure, several frequency bands exhibit higher energy concentrations at specific time intervals, indicated by

the red regions. Unlike MFCC, which compresses feature representations using the Discrete Cosine Transform (DCT), MFE directly preserves the energy values produced by the Mel filter bank, allowing the frequency-domain energy distribution to remain clearly represented. These energy distributions are subsequently used as input features for the neural network model to classify railway sounds and environmental noise.

D. Classification Model Design

The classification model used in this study was developed on the Edge Impulse platform using a Convolutional Neural Network (CNN) architecture. The model was designed to classify audio signals into two categories, namely railway sounds and environmental noise, based on features extracted using the MFCC and MFE methods. CNN was selected because of its ability to automatically learn local patterns from audio feature representations and its proven effectiveness in various audio classification tasks [17], [18].

To ensure a fair comparison between the MFCC and MFE feature extraction methods, the same CNN architecture and training parameters were applied in both experimental scenarios. Consequently, any differences in classification performance can be directly attributed to the feature extraction method rather than variations in the classification model.

TABLE IV

TRAINING MODEL CNN PARAMETER	
Parameter	Value
Epoch	100
Learning Rate	0.005
Validation Set Size	20%
Batch Size	32
Training Processor	CPU
Profile INT8 Model	Aktif

As shown in Table IV, the model was trained for 100 epochs with a learning rate of 0.005, a validation set size of 20%, and a batch size of 32. Training was performed using the CPU available on the Edge Impulse. In addition, the model was configured with the INT8 quantization profile to facilitate deployment on embedded systems with limited memory resources.

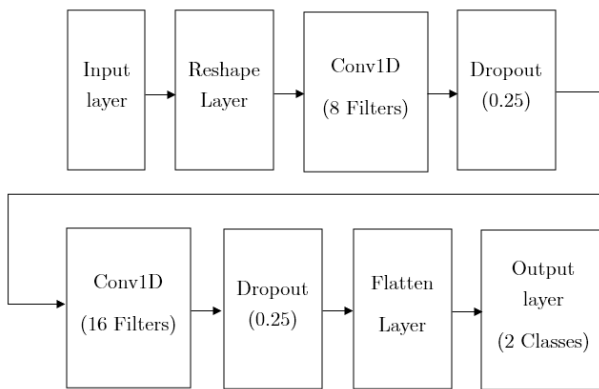


Figure 11. CNN Architecture Used for MFCC and MFE Classification

Figure 11 illustrates the CNN architecture used in this study. The features extracted using the MFCC and MFE methods are

provided as inputs to the input layer. The input data are then processed through two Conv1D layers to learn local patterns from the audio feature representations. A dropout layer with a rate of 0.25 is applied to reduce the risk of overfitting during training, while a flatten layer converts the convolutional feature maps into a one-dimensional vector before classification is performed by the output layer. The model outputs the probability of two classes: railway sounds and environmental noise.

Although both models employ the same CNN architecture, the number of input features differs according to the feature extraction method. The MFCC-based model produces 637 input features, whereas the MFE-based model generates 4,214 input features. This difference arises from the distinct feature representations produced by each extraction method. Nevertheless, the network architecture and all training parameters were kept identical to ensure that the performance evaluation reflected only the effect of the feature extraction method on railway sound classification.

E. System Implementation and Evaluation

Deploying artificial intelligence models on embedded devices enables inference to be performed locally, thereby reducing dependence on external servers [19], [20]. In this study, the trained models were exported and deployed to a Raspberry Pi 5, allowing the system to perform sound classification without requiring an Internet connection or external server. This approach enables autonomous operation with lower latency and reduced network resource consumption [21].

The evaluation was conducted using the testing dataset, which consisted of audio samples that were not used during the training process. The testing dataset included 31 audio samples, comprising 15 railway sound samples and 16 environmental noise samples. Each sample was processed using both feature extraction methods, namely MFCC and MFE, and subsequently classified using the previously trained CNN model.

The performance of the proposed classification system was assessed using a confusion matrix composed of True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN) values. These outcomes were subsequently used to calculate the standard evaluation metrics, namely accuracy, precision, recall, and F1-score, in order to quantify the classification performance of each model [17]. Furthermore, the inference time was recorded to measure the computational efficiency of the deployed models on the Raspberry Pi 5 platform.

To ensure a consistent and unbiased comparison, the CNN models developed using MFCC and MFE features were deployed on the same Raspberry Pi 5 hardware under identical experimental conditions. Their classification performance and execution speed were then analyzed using the predefined evaluation metrics to determine how the selected feature extraction technique influenced the effectiveness of railway sound classification.

III. RESULTS AND DISCUSSION

This section presents the performance of the MFCC- and MFE-based models during both the training stage and deployment on the Raspberry Pi 5. The models were evaluated using accuracy, precision, recall, F1-score, and inference time to investigate the effect of the feature extraction method on the performance of the railway sound classification system.

1) MFCC Model Training Results

The CNN-based classification model using MFCC features was trained on a dataset consisting of 122 training samples and 31 testing samples. Model training was performed on the Edge Impulse using the training parameters described in Section II. The training results of the MFCC-based model are presented in Figure 12.

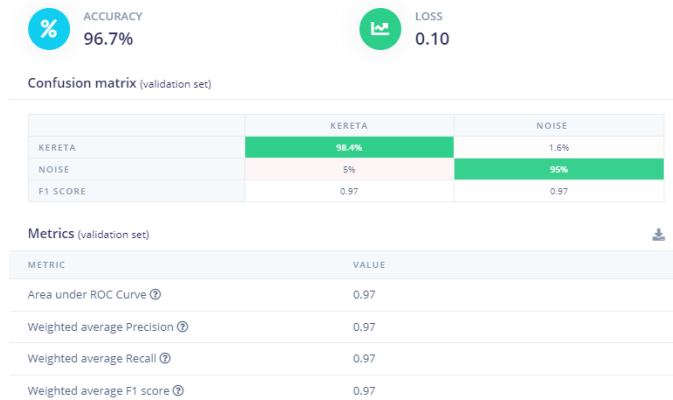


Figure 12. MFCC Training Model Results

As shown in Figure 12, the MFCC-based model achieved a validation accuracy of 96.7% with a validation loss of 0.10. The model also obtained an Area Under the Curve (AUC) of 0.97, a weighted average precision of 0.97, a weighted average recall of 0.97, and a weighted average F1-score of 0.97. These results indicate that the MFCC-based model achieved a high level of classification performance on the validation dataset.

The confusion matrix in Figure 12 shows that the railway sound class was correctly classified at 98.4%, with 1.6% misclassified as environmental noise. Meanwhile, the environmental noise class achieved a correct classification rate of 95.0%, with 5.0% misclassified as railway sound. The F1-score for both the railway sound and environmental noise classes was 0.97, indicating balanced classification performance between the two classes. A visualization of the extracted MFCC feature distribution is presented in Figure 13.

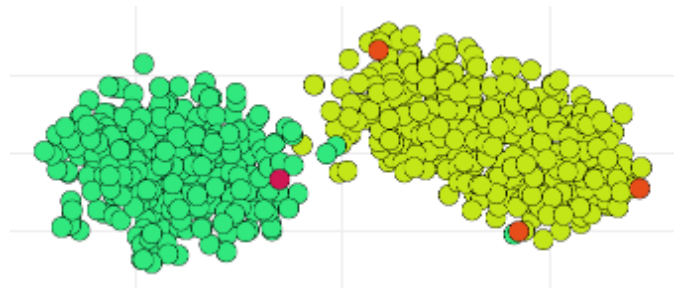


Figure 13. MFCC Feature Distribution in the Data Explorer

Figure 13 presents the distribution of the extracted MFCC features visualized using the Data Explorer in the Edge Impulse platform. Each point represents an individual audio sample after MFCC feature extraction, while the color indicates its corresponding class and classification result. The yellow-green and green points denote correctly classified railway sound and environmental noise samples, respectively, whereas the magenta and orange points indicate misclassified samples. As illustrated in the figure, most railway sound and environmental noise samples form two well-separated clusters. Only a few samples are located near the decision boundary and are misclassified, as indicated by the magenta and orange points. The clear separation between the two clusters demonstrates that MFCC provides an effective representation of the acoustic characteristics of railway sounds and environmental noise, enabling the neural network model to distinguish between the two classes with high classification accuracy.

2) MFE Model Training Results

The CNN model using MFE features was trained using the same dataset as the MFCC-based model to ensure an objective performance comparison. The training results of the MFE-based model on the Edge Impulse platform are presented in Figure 14.

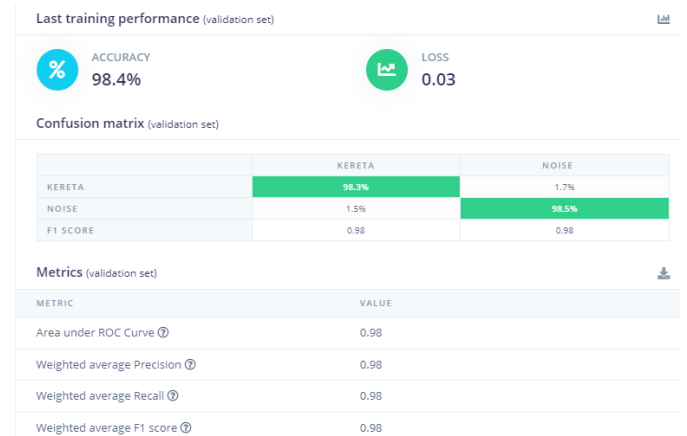


Figure 14. MFE Training Model Results

As shown in Figure 14, the MFE-based model achieved a validation accuracy of 98.4% with a validation loss of 0.03. In addition, the model obtained an Area Under the Curve (AUC) of 0.98, a weighted average precision of 0.98, a weighted average recall of 0.98, and a weighted average F1-score of 0.98.

These results indicate that the MFE-based model achieved high classification performance on the validation dataset.

The confusion matrix shows that the railway sound class was correctly classified at 98.3%, with 1.7% misclassified as environmental noise. Meanwhile, the environmental noise class achieved a correct classification rate of 98.5%, with 1.5% misclassified as railway sound. The F1-score for both the railway sound and environmental noise classes was 0.98, indicating balanced classification performance between the two classes. A visualization of the extracted MFE feature distribution is presented in Figure 15.

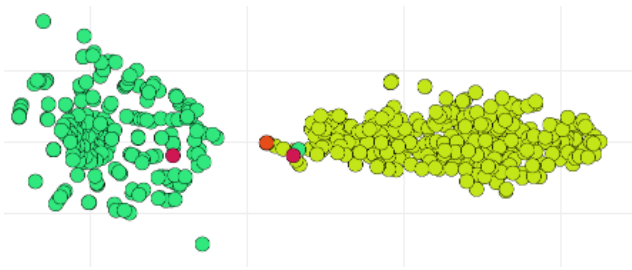


Figure 15. MFE Feature Distribution in the Data Explorer

As shown in Figure 15, the extracted MFE features are visualized using the Data Explorer in the Edge Impulse platform. Each point represents an individual audio sample after MFE feature extraction, while the color indicates its corresponding class and classification result. The yellow-green and green points denote correctly classified railway sound and environmental noise samples, respectively, whereas the magenta and orange points indicate misclassified samples. The feature distribution shows that the MFE representation provides a clear separation between the railway sound and environmental noise classes, although some overlap between the two classes can still be observed. The presence of a small number of misclassified samples indicates that certain audio samples exhibit similar acoustic characteristics between the two classes. This feature representation is consistent with the validation results, where the MFE-based CNN model achieved a validation accuracy of 98.4%, with a weighted precision, recall, and F1-score of 0.98.

3) Model Implementaation on Raspberry Pi 5

After the training process was completed, the MFCC- and MFE-based models were deployed on a Raspberry Pi 5 to perform on-device inference. The deployment used the exported models from the Edge Impulse platform, allowing the classification process to be executed directly on the embedded device without requiring an external server connection. The evaluation was conducted using 31 audio samples, consisting of 15 railway sound samples and 16 environmental noise samples. The performance of both models on the Raspberry Pi 5 is summarized in Table V.

Parameter	MFCC	MFE
Accuracy	96.7742%	100%
Precision	100%	100%
Recall	93.33%	100%
F1-Score	96.55%	100%

As summarized in Table V, the MFCC-based model achieved an accuracy of 96.7742%, with a precision of 100%, recall of 93.33%, and F1-score of 96.55%. In comparison, the MFE-based model achieved an accuracy of 100%, with 100% precision, 100% recall, and 100% F1-score. These results indicate that both feature extraction methods can be successfully deployed on the Raspberry Pi 5 for railway sound classification. The MFE-based model achieved higher classification performance on the testing dataset, while the MFCC-based model also demonstrated high classification performance, with a lower recall resulting from the misclassification of one railway sound sample.

TABLE VI
CONFUSION MATRIX MFCC AND MFE

	MFCC		MFE	
	Railway Sound	Noise	Railway Sound	Noise
Actual Railway Sound	14	1	16	0
Actual Noise	0	16	0	15

As shown in the confusion matrices in Table VI, the MFCC-based model correctly classified 30 out of 31 test samples, with one railway sound sample misclassified as environmental noise. The model correctly classified all 16 environmental noise samples, while 14 of the 15 railway sound samples were correctly identified. In contrast, the MFE-based model correctly classified all 31 test samples, consisting of 15 railway sound samples and 16 environmental noise samples. These results show that the MFE-based model produced no misclassification on the testing dataset, whereas the MFCC-based model had one misclassified railway sound sample.

In addition to classification performance, inference time was measured to evaluate the computational efficiency of each model when deployed on the Raspberry Pi 5. The experimental results show that the MFCC-based model achieved an average inference time of approximately 12.317 ms, whereas the MFE-based model required approximately 23.44 ms. These results indicate that the MFCC-based model performed inference faster than the MFE-based model, while the MFE-based model achieved higher classification performance on the testing dataset.

4) Comparative Analysis of MFCC and MFE

To investigate the effect of the feature extraction method on system performance, a comparative analysis was conducted using the training results obtained from the Edge Impulse platform and the deployment results on the Raspberry Pi 5. The comparison between the MFCC- and MFE-based models is presented in Table VII.

TABLE VII

COMPARISON OF MODEL TRAINING AND DEPLOYMENT RESULTS

Parameter	MFCC Training	MFCC Raspberry Pi 5	MFE Training	MFE Raspberry Pi 5
Accuracy	96.7%	96.7742%	98.4%	100%
Precision	97%	100%	98%	100%
Recall	97%	93.33%	98%	100%
F1-Score	97%	96.55%	98%	100%
Inference Time (ms)	5	12.317	9	23.44

As summarized in Table VII, the MFCC- and MFE-based models achieved high classification performance during the training stage. The MFCC-based model achieved a validation accuracy of 96.7%, with precision, recall, and F1-score of 97%. Meanwhile, the MFE-based model achieved a higher validation accuracy of 98.4%, with precision, recall, and F1-score of 98%. These results indicate that both feature representations enabled the CNN model to distinguish railway sounds from environmental noise effectively, with the MFE-based model achieving higher validation performance.

The deployment results on the Raspberry Pi 5 show that the MFCC-based model achieved an accuracy of 96.7742%, with a precision of 100%, recall of 93.33%, and F1-score of 96.55%. In comparison, the MFE-based model achieved 100% accuracy, precision, recall, and F1-score. These results indicate that the MFE-based model achieved higher classification performance on the testing dataset, while the MFCC-based model also maintained high classification performance with one misclassified railway sound sample.

The difference between the validation and deployment results can be related to the characteristics of the respective datasets. The validation results were obtained during the model training process, whereas the deployment evaluation was conducted using the separate testing dataset on the Raspberry Pi 5. The testing dataset consisted of 31 audio samples, including 15 railway sound samples and 16 environmental noise samples. With this relatively limited number of testing samples, a single misclassification can noticeably affect the resulting accuracy.

In addition to classification performance, computational efficiency was evaluated by measuring the inference time of each model on the Raspberry Pi 5. The MFCC-based model achieved an average inference time of approximately 12.317 ms, whereas the MFE-based model required approximately 23.44 ms. These results show that the MFCC-based model performed inference faster than the MFE-based model on the Raspberry Pi 5.

The difference in inference time is associated with the dimensionality of the extracted feature representations. In this study, the MFCC feature extractor produced 637 input features, whereas the MFE feature extractor generated 4,214 input features. The larger feature representation produced by MFE resulted in a longer inference time on the Raspberry Pi 5. Therefore, the experimental results demonstrate a trade-off between classification performance and computational efficiency: the MFE-based model achieved higher

classification performance, while the MFCC-based model required less inference time on the embedded platform.

5) Real-Time Evaluation of The MFE Model

Since the MFE model achieved the best performance during both the training and Raspberry Pi 5 implementation stages, an additional real-time evaluation was conducted to verify its capability under actual operating conditions. The evaluation was performed near a railway crossing using the complete embedded system consisting of a Raspberry Pi 5, a USB microphone, and a 1.3-inch OLED display. During testing, the OLED displayed the prediction probabilities for the railway sound and environmental noise classes in real time.

TABLE VIII

REAL-TIME EVALUATION RESULT OF THE MFE-BASED MODEL

Test Condition	Railway Sound	Environmental Noise	Prediction
Train Passing 1	96.88%	3.12%	Railway Sound
Train Passing 2	91.41%	8.59%	Railway Sound
Train Passing 3	93.75%	6.25%	Railway Sound
Train Passing 4	93.42%	6.58%	Railway Sound
Train Passing 5	91.76%	8.24%	Railway Sound
Train Passing 6	92.65%	7.35%	Railway Sound
Train Passing 7	94.13%	5.87%	Railway Sound
Train Passing 8	94.27%	5.73%	Railway Sound
Train Passing 9	91.82%	8.18%	Railway Sound
Train Passing 10	92.41%	7.59%	Railway Sound
Environmental noise 1	15.23%	84.77%	Environmental sound
Environmental noise 2	19.14%	80.86%	Environmental sound
Environmental noise 3	4.69%	95.31%	Environmental sound
Environmental noise 4	8.62%	91.38%	Environmental sound
Environmental noise 5	6.83%	93.17%	Environmental sound
Environmental noise 6	7.19%	92.81%	Environmental sound
Environmental noise 7	8.15%	91.86%	Environmental sound
Environmental noise 8	5.48%	94.52%	Environmental sound
Environmental noise 9	6.27%	93.73%	Environmental sound
Environmental noise 10	6.43%	93.57%	Environmental sound

As presented in Table VIII, the deployed MFE model successfully distinguished railway sound from environmental noise during real-time evaluation. In the train-passing condition, the predicted probability of the railway sound class ranged from 91.41% to 96.88%, while the environmental noise probability ranged from 3.12% to 8.59%. In the environmental noise condition, the model produced environmental noise probabilities ranging from 80.86% to 95.31%, while the railway sound probabilities ranged from 4.69% to 19.14%. In all ten train-passing tests, the model predicted the railway sound class, while all ten environmental noise tests were classified as environmental noise. These results demonstrate

that the MFE-based model maintained consistent classification behavior during real-time operation on the Raspberry Pi 5 under the tested environmental conditions.

IV. CONCLUSION

This study successfully compared the MFCC and MFE feature extraction methods for railway sound classification using a CNN model developed on the Edge Impulse platform. The validation results showed that the MFCC-based model achieved an accuracy of 96.7%, with a precision, recall, and F1-score of 97%. Meanwhile, the MFE-based model achieved a higher validation accuracy of 98.4%, with a precision, recall, and F1-score of 98%. The deployment results on the Raspberry Pi 5 showed that the MFCC-based model achieved an accuracy of 96.7742%, with a precision of 100%, recall of 93.33%, and F1-score of 96.55%. In comparison, the MFE-based model achieved 100% accuracy, precision, recall, and F1-score on the testing dataset. Although the MFE-based model required a longer average inference time of 23.44 ms compared with 12.317 ms for the MFCC-based model, the MFE-based model was selected for real-time implementation because it achieved higher classification performance in both validation and testing results. The real-time evaluation further showed that the MFE-based model consistently classified the tested railway sound and environmental noise conditions correctly. These results demonstrate a trade-off between classification performance and computational efficiency, with MFE providing higher classification performance and MFCC providing faster inference on the Raspberry Pi 5.

ACKNOWLEDGEMENTS

The authors would like to express their sincere gratitude to the Department of Electrical Engineering, State Polytechnic of Malang, for providing academic guidance and research facilities throughout this study. The authors also gratefully acknowledge the financial support provided through the DIPASwadana State Polytechnic of Malang 2026, which significantly contributed to the successful completion of this research.

REFERENCES

- [1] A. Guzhov, F. Raue, J. Hees, and A. Dengel, "ESResNet: Environmental Sound Classification Based on Visual Domain Models," in *Proc. Int. Conf. Pattern Recognition (ICPR)*, 2020, pp. 492–499.
- [2] J. Sharma, O. C. Granmo, and M. Goodwin, "Environment Sound Classification Using Multiple Feature Channels and Attention Based Deep Convolutional Neural Network," *arXiv preprint arXiv:1908.11219*, 2019.
- [3] S. Hershey et al., "CNN Architectures for Large-Scale Audio Classification," in *Proc. IEEE Int. Conf. Acoustics, Speech and Signal Processing (ICASSP)*, 2017, pp. 131–135.
- [4] D. Stowell and M. D. Plumbley, "Automatic Large-Scale Classification of Bird Sounds is Strongly Improved by Unsupervised Feature Learning," *PeerJ*, vol. 2, p. e488, 2014.
- [5] E. Fonseca, X. Favory, J. Pons, F. Font, and X. Serra, "FSD50K: An Open Dataset of Human-Labeled Sound Events," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 30, pp. 829–852, 2022.
- [6] B. Zhu, K. Xu, D. Wang, L. Zhang, B. Li, and Y. Peng, "Environmental Sound Classification Based on Multi-Temporal Resolution Convolutional Neural Network Combining with Multi-Level Features," *IEEE Access*, vol. 6, pp. 51412–51420, 2018.
- [7] J. Salamon, C. Jacoby, and J. P. Bello, "Deep Convolutional Neural Networks and Data Augmentation for Environmental Sound Classification," *IEEE Signal Process. Lett.*, vol. 24, no. 3, pp. 279–283, 2017.
- [8] Y. Tokozume, Y. Ushiku, and T. Harada, "Learning from Between-Class Examples for Deep Sound Recognition," in *Proc. Int. Conf. Learning Representations (ICLR)*, 2018.
- [9] M. Cartwright, B. Cramer, J. Salamon, O. Nov, and J. P. Bello, "SONYC Urban Sound Tagging (SONYC-UST): A Multilabel Dataset from an Urban Acoustic Sensor Network," in *Proc. IEEE Workshop Applications Signal Processing to Audio and Acoustics (WASPAA)*, 2019, pp. 35–39.
- [10] Y. Su, K. Zhang, J. Wang, and K. Madani, "Environmental Sound Classification Using a Two-Stream CNN Based on Decision-Level Fusion," *Sensors*, vol. 19, no. 7, p. 1733, 2019.
- [11] S. M. Abdoli, P. Cardinal, and A. L. Koerich, "End-to-End Environmental Sound Classification Using a 1D Convolutional Neural Network," *Expert Syst. Appl.*, vol. 136, pp. 252–263, 2019.
- [12] M. A. Ferrag, L. Maglaras, and A. Ahmim, "Audio Classification and Sound Event Detection: A Survey," *IEEE Access*, vol. 11, pp. 13487–13525, 2023.
- [13] N. Zeghidour, O. Teboul, F. de Chaumont Quitry, and M. Tagliasacchi, "LEAF: A Learnable Frontend for Audio Classification," *ICLR*, 2021.
- [14] R. Ardiansyah, B. Bastiar, Adzikirani, D. Marya, and A. Novianti, "Optimasi Penerjemahan Bahasa Asing dengan Teknologi IoT pada Kelas Internasional Politeknik Negeri Malang," *Jurnal ELTEK*, vol. 23, no. 1, pp. 1–8, Apr. 2025.
- [15] D. Chicco and G. Jurman, "The Advantages of the Matthews Correlation Coefficient (MCC) over F1 Score and Accuracy in Binary Classification Evaluation," *BMC Genomics*, vol. 21, no. 1, pp. 1–13, 2020.

- [16] S. Davis and P. Mermelstein, "Comparison of Parametric Representations for Monosyllabic Word Recognition in Continuously Spoken Sentences," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 28, no. 4, pp. 357–366, 1980
- [17] M. Ravanelli and Y. Bengio, "Speaker Recognition from Raw Waveform with SincNet," in *Proc. IEEE Spoken Language Technology Workshop (SLT)*, 2018, pp. 1021–1028.
- [18] K. J. Piczak, "Environmental Sound Classification with Convolutional Neural Networks," in *Proc. IEEE Int. Workshop Machine Learning for Signal Processing (MLSP)*, 2015, pp. 1–6.
- [19] R. A. Asmara, M. Ridwan, G. B. P., and A. N. Handayani, "Pengembangan Sistem Face Recognition Menggunakan Cloud Service, Raspberry Pi dan Convolutional Neural Network (CNN)," *Jurnal ELTEK*, pp. 95–102, 2022.
- [20] D. Aldiani, G. Dwilestari, H. Susana, R. Hamonangan, and D. Pratama, "Implementasi Algoritma CNN dalam Sistem Absensi Berbasis Pengenalan Wajah," *Jurnal ELTEK*, pp. 197–202, 2024.
- [21] V. Tarigan, A. Yusupa, and R. Syahputra, "Optimasi Deteksi Penyakit Alzheimer dengan Convolutional Neural Network (CNN) dan Support Vector Machine (SVM) untuk Klasifikasi Tingkat Demensia," *Jurnal ELTEK*, pp. 453–460, 2025.