Leo, D. and Muhammad, A. H. (no date) "Music Genre Classification Based on Spectrogram Using CNN-MobileNet", Sebatik, 29(2).
Submitted: 2025-07-15 Accepted: 2025-08-18 Published: 2025-12-26

Sebatik Vol. 29 No. 2 December 2025
ISSN: 1410-3737(p) 2621-069X(e)

DOI:10.46984/sebatik.v29i1.2634

Music Genre Classification Based on Spectrogram Using CNN-MobileNet

Donatus Leo[1)], Alva Hendi Muhammad[2)]

[1,2] PJJ S2 Informatika, Universitas Amikom
[1,2,] Jl. Ring Road Utara, Condong Catur, Sleman, Yogyakarta
E-mail: donatus.leo@students.amikom.ac.id[1)], alva@amikom.ac.id[2)]

ABSTRACT

Music is a universal form of art that has a significant impact on human life. In the digital era, managing increasingly large music collections requires an effective classification system to facilitate searching and storage. One of the growing methods is music genre classification, which helps organize music based on specific characteristics. This study explores the application of Convolutional Neural Network (CNN) and the MobileNet architecture for music genre classification based on spectrogram images. Spectrogram representation is used to convert audio signals into visual form, allowing the classification problem to be approached as an image classification task. The dataset used is GTZAN, consisting of six genres: blues, classical, country, hiphop, jazz, and metal. Image augmentation is applied to increase the diversity of training data, including rotation, translation, zooming, brightness adjustment, and horizontal flipping. The evaluation results show that the CNN-MobileNet model achieves an overall accuracy of 83%, with a macro precision of 85%, macro recall of 83%, and macro F1-score of 84%. The classical genre achieved the best performance with an F1-score of 93%. This research demonstrates that spectrogram-based music genre classification using CNN-MobileNet is an effective approach for automatic music recognition tasks

***Keywords**: CNN, Deep Learning, MobileNet, Music Classification, Spectrogram*

1. INTRODUCTION

Music is a universal art form and has a significant impact on human life. Music plays a crucial role in the cognitive and emotional development of humans, especially children. In the digital world, managing ever-growing music collections requires an effective classification system to facilitate searching and storage. One emerging method is music genre classification (Tzanetakis & Cook, 2002), which helps organize music based on specific characteristics. However, the main challenge in music genre classification is the complexity of the music itself, particularly using spectrogram images, which tend to have structures similar to those of other music genres (Sridhar, 2024).

Music genres have complex acoustic and structural characteristics, necessitating automated methods to classify them with high accuracy. Spectrogram image representation allows audio signals to be converted into visual forms, enabling the application of deep learning-based image classification methods (Li, 2024). Although spectrograms are images, the differences between genres cannot be visually identified by humans, necessitating artificial intelligence techniques. In recent years, Convolutional Neural Networks (CNN) have emerged as one of the most effective techniques for various pattern recognition tasks, including music genre classification (Alzubaidi, dkk, 2021). CNNs have the ability to automatically extract complex features from input data, making them highly suitable for handling a variety of audio data in the form of images, including image classification in general (Reza Fahcruroji, dkk, 2024).

Early research on music genre classification often used manual feature-based methods such as the Mel-Frequency Cepstral Coefficient (MFCC). (Yehezkiel & Suyanto, 2022) were among the pioneers in using this technique to classify audio signals. However, manual feature-based approaches often have limitations because they cannot fully capture the complex representations present in music. Therefore, developments in artificial intelligence technology, particularly deep learning, have opened new opportunities to address this challenge.

The use of CNN models has become a popular approach in music genre classification due to its ability to analyze visual data such as spectrograms (Purnama, 2022). CNN architectures such as ResNet-50 and VGG-16 can perform well in spectrogram analysis for music genre recommendation. This suggests that visual representations of audio, such as spectrograms, can be used to more effectively capture the characteristics of music genres.

In comparison, other hybrid approaches that combine CNN with RNN variants such as Gated Recurrent Unit (GRU) and Bidirectional GRU (Bi-GRU) have also shown significant results. The CNN-BiGRU architecture, using spectrogram features, achieved a peak accuracy of 89.30%, significantly better than conventional methods (Ashraf, dkk, 2023). This confirms that the combination of hybrid models can provide a more dynamic and accurate solution for music genre classification. Similarly, research by Dutta & Chanda (2024) revealed that CNN-LSTM can improve music classification accuracy compared to pure CNN (Dutta & Chanda, 2024).

In addition to deep learning models, research (Falola, dkk, 2022) reviews the development of machine learning and deep learning techniques for music genre classification. The study highlights the importance of selecting appropriate input features, such as spectrograms and MFCC, in improving model performance. This research also emphasizes the need for a more flexible approach to handle various music genres. Meanwhile, in Indonesia, research on music genre classification is still developing, successfully classifying Indonesian music genres using CNNs (Wairata, dkk, 2021).

This study aims to develop a CNN model with the MobileNet architecture (Andrew G. Howard, 2017) as a lightweight and efficient architecture for implementing spectrogram-based music genre classification for blues, classical, country, hip-hop, jazz, and metal using the GTZAN dataset. Although existing research on music classification has focused limitedly on spectrogram images combining CNNs with MobileNet. This approach is expected to serve as a reference for further model development.

2. FOCUS AND SCOPE

This research focuses on the development and evaluation of a music genre classification model using a deep learning approach based on spectrogram images. The data used is the GTZAN Genre Collection dataset, which consists of six music genre categories. Each audio file is converted into a spectrogram image. The two main approaches used in this study are a pure CNN and a CNN combined with MobileNet as a transfer learning model. The model training and testing process was conducted using the Python programming language with the TensorFlow/Keras library on Google Colab. Model performance evaluation was based on precision, recall, and f1-score metrics to measure the classification effectiveness of each model. The limitations of this study are:

1. This study only used six music genres out of the original 10 in the GTZAN dataset to address performance issues during model training due to device limitations.
2. The classification process was conducted based on the converted spectrogram images, not directly on audio files.
3. The focus was on the CNN architecture and MobileNet as a feature extractor.
4. The model was built and evaluated on the Google Colab platform, which has limited hardware (RAM and GPU).
5. This research does not discuss audio segmentation techniques, mixed genres (multi-label), or real-time classification.

This research is expected to produce a high-performance music genre classification model for genre recognition based on spectrogram images, using both pure CNN and CNN-MobileNet. Furthermore, this research is also expected to provide a comparative overview of the effectiveness of transfer learning in improving model training accuracy and efficiency. The results of this study are expected to serve as an initial reference for the development of a broader automatic music classification system and contribute to the application of deep learning in the fields of digital audio and music.

3. MATERIALS AND METHODS

This research is an experimental study aimed at testing the performance of a Convolutional Neural Network (CNN) model in music genre classification based on spectrograms (Fardhan, dkk, 2021). This research is quantitative in nature, employing computational and machine learning approaches. Music data will be converted into spectrograms for analysis using CNN.

3.1 Dataset

This research uses the GTZAN dataset (Fardhani dkk, 2021), which consists of 10 popular music genres. However, this study only used 6 music genres to maximize performance during model training. The genres used in this study are: blues, classical, country, hip-hop, jazz, and metal. The total data consists of 600 spectrogram images, each genre having 100 spectrogram images, which are divided into 70 training data, 15 validation data, and 15 test data, as shown in Table 1. This dataset consists of spectrogram images, as can be seen in Figure 1.

Table 1. Dataset distribution

No	Gendre	Train	Val	Test
1	Blues	70	15	15
2	Classical	70	15	15
3	Country	70	15	15
4	Hiphop	70	15	15
5	Jazz	70	15	15
6	Metal	70	15	15

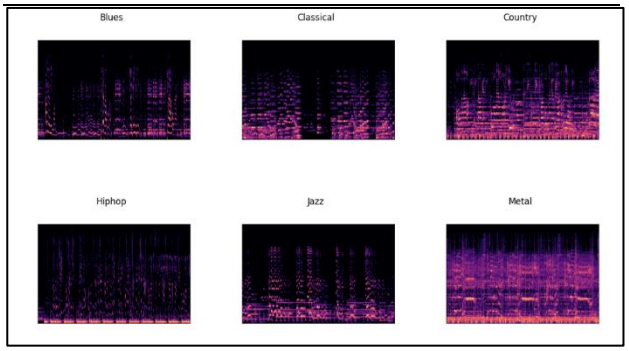

Figure 1. Spectrogram

3.2 Preprocessing and Augmentation

At this stage, data preprocessing was performed to ensure that all spectrogram images (Sridhar, 2024) had uniform dimensions and standardized pixel values. All

images were rescaled to 128×128 pixels using the IMG_HEIGHT and IMG_WIDTH parameters, and processed in batches of 32 images (BATCH_SIZE = 32). This aimed to optimize the model training process and GPU memory usage.

For the training data, a real-time augmentation technique was used using the ImageDataGenerator class from TensorFlow. This augmentation technique aims to increase data diversity without manually increasing the amount of data and to help the model generalize better to new data. Some of the transformations applied are shown in Table 2.

Table 2. Preprocessing and Augmentation

Parameter	Value	Description
Rescale	1./255	Normalize pixel values from the range 0–255 to 0–1.
Rotation_range	20	Random rotation up to 20 degrees, preserving the visual structure of the spectrogram.
Width_shift_range	0.15	Horizontal shift of 15% of the image width.
Height_shift_range	0.15	Vertical shift of 15% of the image height.
Zoom_range	0.2	Random zoom up to 20% to vary the scale of features.
Shear_range	15	Oblique distortion up to 15 degrees, adding variation to the shape of features.
Brightness_range	[0.7, 1.3]	Illumination variation to reflect different lighting conditions.
Horizontal_flip	TRUE	Horizontal flip, experimental to detect possible symmetry in the spectrogram.
Fill_mode	'nearest'	Filling empty areas after transformation with the nearest pixel value.

No augmentation was applied to the validation and testing data. The images were simply normalized (rescale=1./255) to ensure the model consistently evaluates the data, without any additional modifications that could affect the prediction results. The dataset was loaded using the flow_from_directory() function, which reads the folder structure based on class labels. Three generators were created, one each for training (train_generator), validation (val_generator), and testing (test_generator) with the class_mode='categorical' parameter to support multiclass classification, and shuffle was set only on the training data to maintain a random distribution.

3.3 Convolutional Neural Networks (CNN)

Convolutional Neural Networks (CNN) are one of the most popular Deep Learning networks used for image recognition (Alzubaidi, dkk, 2021). The following is a CNN architecture for image recognition in Figure 2.

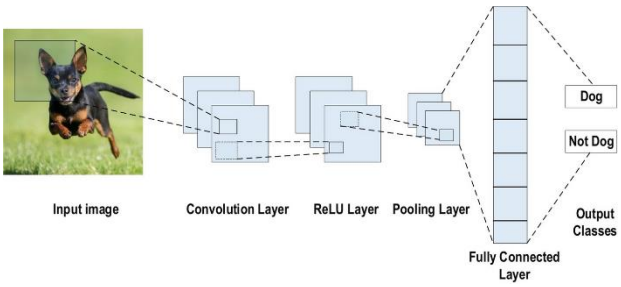

Figure 2. An example of CNN architecture for image classification

CNNs are also popular in Indonesia for image classification, such as shadow puppets (Khoirun Nisa' & Riadi, 2025), and for identifying plant diseases from leaf images (Asrafil, dkk, 2020).

This study will attempt to build a model using two approaches: a conventional CNN and a CNN with Mobilenet.

1. CNN Model

The initial model used by this researcher was a standard CNN model with a sequential model display as shown in Figure 3.

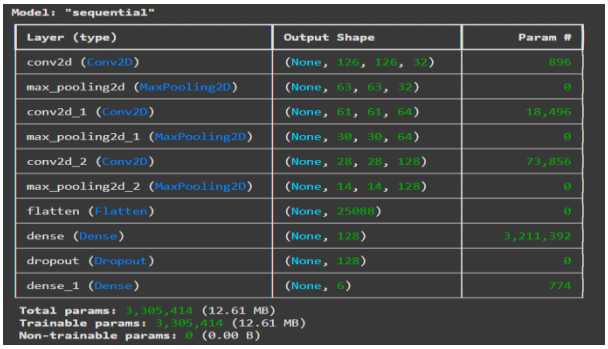

Figure 3. CNN Sequential Model

The sequential model in Figure 3 represents a Convolutional Neural Network (CNN) architecture used to classify images into six classes. This model consists of three Conv2D and MaxPooling2D blocks, each tasked with extracting spatial features from the spectrogram image and reducing its spatial dimensionality. Each convolutional layer uses an increasing number of filters (32, 64, 128), indicating a deepening of the feature representation. The results are then flattened through a Flatten layer into a one-dimensional vector, which is then

processed by a Dense layer with 128 neuron units and dropout to reduce overfitting. Finally, a final Dense layer with six neurons and a softmax activation function is used to generate probability predictions for six music genre classes. This model has a total of 3,305,414 trainable parameters, demonstrating considerable learning capacity. The stages of this CNN modeling can be seen in Table 3.

Table 3. CNN Model

No	Layer	Main Function	Main Parameters
1	Conv2D (32, (3,3), activation ='relu')	Initial local feature extraction (edges, lines)	Filter: 32, Kernel: 3x3, Aktivasi: ReLU
2	MaxPooling2D (2,2)	Dimensionality reduction and overfitting	Pool Size: 2x2
3	Conv2D (64, (3,3), activation ='relu')	Intermediate feature extraction (shape patterns)	Filter: 64, Kernel: 3x3, Aktivasi: ReLU
4	MaxPooling2D (2,2)	Advanced feature reduction	Pool Size: 2x2
5	Conv2D (128, (3,3), activation ='relu')	High-level feature extraction (complex structures in the spectrogram)	Filter: 128, Kernel: 3x3, Aktivasi: ReLU
6	MaxPooling2D (2,2)	Simplifying the final feature representation	Pool Size: 2x2
7	Flatten ()	Converting 3D tensors to 1D	-
8	Dense (128, activation ='relu')	Combining all features into a classification representation space	Neuron: 128, Aktivasi: ReLU
9	Dropou t (0.5)	Regularization to reduce overfitting	Rasio dropout: 50%
10	Dense(len (label_map), activation ='softmax')	Classification into music genre classes	Neuron: according to the number of classes, Activation: Softmax

This CNN model consists of several layers that play a crucial role in the feature extraction and classification process. The first layer is a Conv2D layer with 32 3x3 filters and a ReLU activation function, which captures local patterns such as edges or lines in the spectrogram image. ReLU activation is crucial for adding non-linearity so the model can learn more complex patterns. Next, the convolution results are processed by a 2x2 MaxPooling2D layer, which extracts the maximum value from each 2x2 block, reducing the feature size, increasing computational efficiency, and reducing the risk of overfitting without losing important features.

The second convolution layer (Conv2D) increases the number of filters to 64, allowing the model to capture more in-depth information such as the distinctive shapes or patterns of a particular music genre. This process is followed by a second pooling process, which maintains efficiency while retaining important information. The third convolution layer adds 128 filters and is used to extract more abstract, high-level features, such as the unique frequency structure of a particular music genre. Afterward, a final 2D MaxPooling is used to simplify the extraction results before entering the classification stage.

After the feature extraction process is complete, the output in the form of a 3D tensor is converted into a 1-dimensional vector through a Flatten layer, as the first step to entering the classification phase. This vector is then processed by a fully connected (Dense) layer with 128 neurons and a ReLU activation function, which combines all previously extracted feature information. To prevent overfitting, a Dropout layer with a ratio of 0.5 is applied, where half the neurons are randomly deactivated during training. Finally, a Dense output layer with the number of neurons corresponding to the number of genre classes and softmax activation is used to generate probabilities for each class, with the class with the highest probability being the model's prediction.

2. CNN-MobileNet Model

The next model in this study will use the Mobilenet architecture within the CNN model itself (Andrew G. Howard, 2017). The visual representation of the CNN model with Mobilenet can be seen in Figure 4.

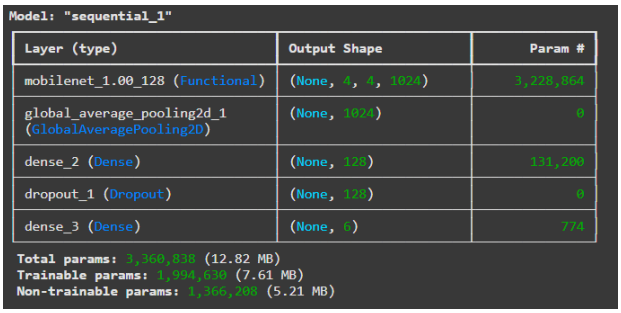

Figure 4. CNN-Mobilenet Sequential Model

This model is a convolutional neural network (CNN) architecture that uses transfer learning from MobileNetV1 with a width multiplier of 1.00 and an input resolution of 128 pixels. The model consists of several main layers. The first layer, mobilenet_1.00_128, acts as a feature extractor, extracting important features from the spectrogram image and producing an output of size (4, 4, 1024). This layer contributes the majority of parameters, approximately 3,228,864, but most of them are non-trainable (1,366,208 parameters), meaning the weights are not updated during the training process because the model uses pretrained weights.

Next, the output from MobileNet is passed to the GlobalAveragePooling2D layer, which flattens the spatial features into a single 1024-dimensional vector, simplifying complexity without losing representation of important features. This vector is then passed to a Dense layer with 128 neurons (using ReLU activation) containing 131,200 trainable parameters, tasked with performing a non-linear transformation on the features to better prepare them for classification. Next, there's a Dropout layer with 128 units that serves as regularization to prevent overfitting. Finally, the output layer is a Dense layer with 6 neurons (the number of music genre classes), which represents the final predictions in the form of probabilities for each class. Overall, this model has a total of 3,360,838 parameters, of which 1,994,630 are trainable and the rest are non-trainable. This indicates that the transfer learning strategy successfully reduces computational requirements and training time without sacrificing performance. This model is well-suited for the task of spectrogram-based music genre classification because it combines the feature extraction advantages of MobileNet with the adaptability of additional dense layers. The flow of this model can be seen in Table 4.

Table 4. CNN-MobileNet Model

No	Layer	Main Function	Main Parameters
1	MobileNet (without top layer)	Extracting initial features from spectrogram images with pretrained weights from ImageNet	include_top=False, weights='imagenet', input_shape=(IMG_HEIGHT, IMG_WIDTH, 3)
2	GlobalAveragePooling2D	Flattening spatial features from MobileNet output into a 1-dimensional vector	-
3	Dense	Adding complexity to feature representation with ReLU activation	units=128, activation='relu'
4	Dropout	Reducing overfitting by randomly deactivating neurons during training	rate=0.5
5	Dense (Output Layer)	Performing final classification into several music genre classes	units=number_of_classes, activation='softmax'

The model begins with the first layer, MobileNet, used without a final classification section (include_top=False). This layer acts as the primary feature extractor, extracting important visual patterns from spectrogram images using weights pre-trained on the ImageNet dataset. With input images of size (IMG_HEIGHT, IMG_WIDTH, 3), MobileNet processes the data using a series of efficient depthwise separable convolution layers, making it very lightweight yet powerful in capturing features.

The extracted features from MobileNet are then passed to the GlobalAveragePooling2D layer, which flattens the spatial features into a single-dimensional vector. This layer replaces the traditional flatten layer because it is more efficient and empirically helps reduce overfitting. Once the feature vector is obtained, the next layer is a Dense layer with 128 units and a ReLU activation function. This layer adds non-linear complexity to the data representation, allowing the model to more clearly distinguish between classes.

To prevent the model from over-relying on certain features (overfitting), a Dropout layer with a dropout ratio of 0.5 is applied. This layer randomly deactivates 50% of neurons during training, making the model more robust to data variations. Finally, there's a Dense output layer with the same number of units as the number of music genre classes, using the softmax activation function. The softmax function generates a probability distribution for each class, allowing the model to determine the final genre prediction from a given spectrogram input.

With this combination of structures, the model combines the power of MobileNet in feature extraction with the flexibility of additional dense and dropout layers to accurately and efficiently classify music genres.

3.4 Model Training with EarlyStopping

The model training process was performed using the model.fit() function from the Keras library, implementing the EarlyStopping strategy (Andika Surya, dkk, 2025) to prevent overfitting and maintain model generalization. This can be seen in Figure 5.

```
from tensorflow.keras.callbacks import EarlyStopping

early_stop = EarlyStopping(
    monitor='val_loss',
    patience=5,
    restore_best_weights=True
)

history = model.fit(
    train_generator,
    epochs=30,
    validation_data=val_generator,
    callbacks=[early_stop]
)
```

Figure 5. Train Model with EarlyStopping

Training data is supplied through the train_generator object, while validation data is provided by the

val_generator. Training is designed to run for a maximum of 30 epochs, but can be stopped early if model performance on the validation data shows no improvement.

The EarlyStopping technique is used by monitoring the validation loss (val_loss) value at each epoch. The patience parameter is set to 5, meaning training will automatically stop if there is no decrease in the val_loss value for five consecutive epochs. Furthermore, the restore_best_weights parameter is set to True, allowing the model to restore the best weights (i.e., those with the lowest val_loss) after the training process is complete. With this strategy, the resulting model is expected to have optimal performance and not experience degradation due to overtraining.

The entire training history, including loss and accuracy values on both training and validation data, is recorded in the history variable. This data can later be used for further model performance analysis, both numerically and visually through training graphs.

3.5 Model Evaluation

To evaluate the performance of the music genre classification model, three main metrics were used: precision, recall, and F1-score.

$$Precision = \frac{TP}{TP+FP} \quad (1)$$

$$Recall = \frac{TP}{TP+FN} \quad (2)$$

$$F1\ Score = 2x\frac{Precision\ x\ Recall}{Precision + Recall} \quad (3)$$

Precision measures the proportion of correct predictions relative to all predictions for a class (1). Recall measures the model's ability to recognize all instances of a class (2). The F1-score is the harmonic mean of precision and recall, used to balance the two, especially when data imbalance occurs (3). TP stands for True Positive, FP stands for False Positive, and FN stands for False Negative.

The model evaluation process was conducted to assess the performance of conventional CNN architectures and the combination of CNN and MobileNet in spectrogram-based music genre classification. This evaluation used validation and test data, with reference to several performance metrics, including accuracy, loss, precision, recall, and F1-score. Additionally, graphs of accuracy and loss versus the number of epochs were used to visualize the learning process and model stability during training.

To maintain training accuracy and efficiency, an early stopping technique was implemented, automatically halting the training process when performance on the validation data no longer improves. This evaluation stage was designed to ensure that the developed model not only learns from the training data but also has good generalization capabilities to previously unseen data. All evaluation steps were carried out systematically and consistently on both model architectures tested.

3.6 Flowchart of Research Methodology

This research involves several steps and processes, as shown in Figure 6. It begins with dataset collection, followed by pre-processing and data augmentation to improve the quality and diversity of the input. Next, the model building process is carried out, where two architectures are tested: a standard CNN model and a CNN model combined with MobileNet. Both models are then trained using an early stopping technique to prevent overfitting. After training is complete, the final stage is model evaluation to assess the resulting classification performance.

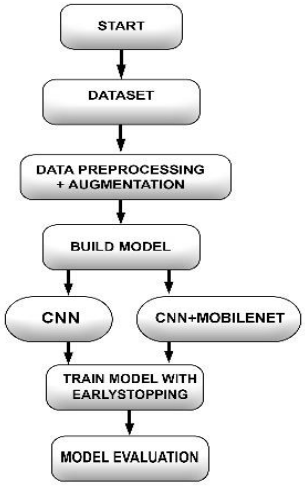

Figure 6. Research flow

4. RESULTS AND DISCUSSION

This section provides a comprehensive discussion and in-depth analysis of the performance of the two spectrogram image classification models developed in this study: a conventional Convolutional Neural Network (CNN) and a CNN enhanced with the MobileNet architecture. These two models were chosen to explore the comparative strengths of a basic CNN framework versus a more optimized, lightweight, and mobile-friendly architecture like MobileNet, especially when dealing with the complexities of spectrogram-based music genre classification.

The evaluation is carried out by systematically comparing the results obtained during the training and testing phases for both models. These comparisons focus on multiple key performance metrics, including accuracy and loss progression over epochs, as presented in Figure 7. In addition, confusion matrix visualizations are analyzed to observe how each model performs in correctly and incorrectly classifying individual music genres. These matrices provide a clearer picture of the distribution of classification results and help highlight patterns of misclassification that may be occurring due to similarities in spectral features among different genres.

Furthermore, a detailed comparison is conducted using precision, recall, and F1-score values for each class. These metrics are particularly valuable in assessing the reliability and robustness of each model beyond overall accuracy. For instance, precision helps measure the ability of the model to avoid false positives, recall reflects how well the model captures all relevant instances, and the F1-score provides a harmonic mean that balances both precision and recall. By analyzing these metrics, the study aims to deliver a nuanced understanding of how well each model performs across various genre categories.

This discussion is intended not only to highlight the effectiveness of the models in learning and recognizing the unique characteristics present in spectrogram representations of audio data but also to identify potential limitations or biases in the classification process. Through this comprehensive evaluation, it becomes possible to assess which model architecture offers better generalization capability, higher computational efficiency, and greater suitability for real-world applications, such as mobile music recognition systems or automated playlist categorization.

Ultimately, the insights gained from comparing these two architectures provide valuable implications for future work in the domain of audio signal processing and machine learning, particularly in optimizing model selection for specific classification tasks. The accuracy and loss graphs depicted in Figure 7 serve as the foundation for this analysis, offering a visual representation of the models' learning behavior throughout the training process.

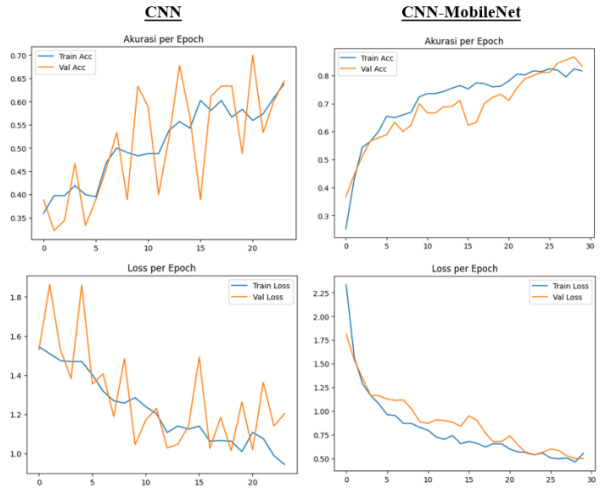

Figure 7. Accuracy and loss graph

Figure 7 shows a comparison of the performance of a conventional CNN model and a CNN combined with the MobileNet architecture based on accuracy and loss metrics during the training and validation processes.

In the conventional CNN model, the accuracy graph shows a gradual increase from around 0.38 to nearly 0.65 at the end of the 22nd epoch. However, the validation accuracy appears to fluctuate and tends to be unstable, indicating a possible overfitting issue or suboptimal model generalization to the validation data. This is reinforced by the inconsistent trend between the training and validation accuracy. In terms of loss, the training loss value consistently decreases, but the validation loss shows a sharp up-and-down pattern. This condition confirms the indication that the model is still struggling to consistently capture generalization patterns from the validation data.

In contrast, the CNN model with the MobileNet architecture as a feature extractor demonstrates much better and more stable performance. Training and validation accuracy have increased significantly and consistently since the initial epoch, with validation accuracy reaching above 0.85 at the end of the 30th epoch.

The parallel pattern of accuracy increases between training and validation indicates that the model has good generalization capabilities. Furthermore, the loss graph for this model shows a sharp and steady decrease in both the training and validation data. This indicates that the optimization process is more effective with the CNN-MobileNet model than with the conventional CNN.

The next evaluation will be presented in the form of a Confusion Matrix visualization, as shown in Figure 8.

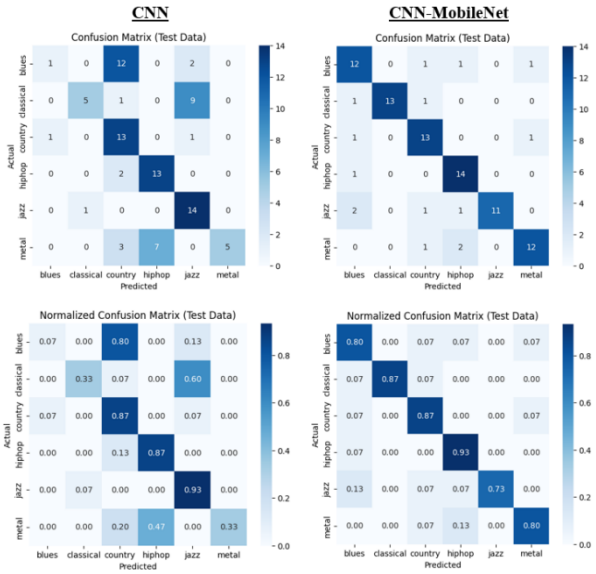

Figure 8. Confusion matrix

Figure 8 shows a confusion matrix comparing the performance of the pure CNN model and the CNN-MobileNet model in classifying six music genres. The pure CNN model's classification performance remains unstable. Some classes, such as jazz (93%) and hip-hop (87%), did achieve high accuracy. However, most other classes exhibited relatively high misclassification rates. For example, only 33% of the classical class data was correctly classified, while the majority of the remaining data was predicted as jazz. Similarly, the metal class was only correctly recognized 33% of the time, with most

predictions pointing to hip-hop and country. This indicates that the pure CNN model struggled to distinguish between several music genres with similar spectral characteristics, resulting in high levels of confusion between the classes.

In contrast, the CNN model using the MobileNet architecture significantly improved classification performance. Nearly all classes demonstrated higher prediction accuracy and a more balanced error distribution. The classical, country, and hip-hop classes were each correctly classified 87% of the time, while metal was successfully recognized with 80% accuracy. Although the jazz class showed a slight decrease in accuracy (73%), the prediction errors were not concentrated in a single class, but rather spread across several other classes, such as blues, country, and hip-hop. This indicates that the MobileNet model has better generalization capabilities in distinguishing complex patterns across genres, resulting in more stable and accurate performance overall compared to conventional CNNs.

For a detailed discussion, please see the following comparative table 5.

Table 5. Comparator

Genre / Metrik	Precision (CNN)	Recall (CNN)	F1-Score (CNN)	Precision (CNN+ MobileNet)	Recall (CNN+ MobileNet)	F1-Score (CNN+ MobileNet)
Blues	0.50	0.07	0.12	**0.71**	**0.80**	**0.75**
Classical	0.83	0.33	0.48	**1.00**	**0.87**	**0.93**
Country	0.42	0.87	0.57	**0.76**	**0.87**	**0.81**
HipHop	0.65	0.87	0.74	**0.78**	**0.93**	**0.85**
Jazz	0.54	0.93	0.68	**1.00**	**0.73**	**0.85**
Metal	1.00	0.33	0.50	**0.86**	**0.80**	**0.83**
Akurasi Keseluruhan	**0.57**	–	–	**0.83**	–	–
Macro Average	0.66	0.57	0.51	**0.85**	**0.83**	**0.84**
Weighted Average	0.66	0.57	0.51	**0.85**	**0.83**	**0.84**

Table 5 above presents a performance comparison between the CNN and CNN-MobileNet models on the spectrogram image-based music genre classification task. Overall, the CNN-MobileNet model demonstrated significant and consistent performance improvements across all evaluation metrics, namely precision, recall, and F1-score. One of the most striking improvements occurred in the blues genre, where the F1-score increased sharply from 12% to 75%. This indicates that the original CNN model failed to recognize patterns in that genre, while the MobileNet architecture was able to capture more relevant features. In the metal genre, although the CNN model achieved perfect precision (100%), its recall was very low (33%), indicating that much of the data was not recognized. In contrast, the combined CNN-MobileNet model performed more balanced and realistically, with an F1-score of 83%.

Furthermore, an improvement was also seen in the macro-average F1-score, which rose from 51% to 84%. This indicates that the performance improvement occurred evenly across all classes, not just focused on one or two genres. The overall accuracy on the test data also increased from 57% to 83%, reflecting a much better generalization ability than the conventional CNN model.

5. CONCLUSION

Based on the evaluation results, it can be concluded that the integration of the MobileNet architecture into the CNN model significantly improves the performance of spectrogram-based music genre classification. This combined model shows consistent improvements in key evaluation metrics such as precision, recall, and F1-score, and provides higher accuracy and better generalization compared to conventional CNNs. The sharp improvement in certain genres, such as blues and metal, demonstrates that the CNN-MobileNet model is able to recognize features that were previously missed by the baseline model. Overall, the use of MobileNet as a feature extractor has proven effective in strengthening the model's ability to understand the complexity of spectrogram data in the context of music genre classification.

6. SUGGESTIONS

For further research, it is recommended that the model be further studied using more complex or adaptive deep learning architectures, such as EfficientNet or ResNet, to explore the potential for greater accuracy improvements. Furthermore, testing on larger and more diverse datasets is necessary to enhance the model's generalizability and ability to recognize genres with similar spectrogram characteristics. Further augmentation techniques, such as time masking or frequency masking, can also be explored to strengthen the model's robustness to data variations. Finally, integrating this model into real-world applications, such as music recommendation systems or automatic classification in digital audio platforms, is a potential direction for further development.

7. REFERENCE

Alzubaidi, L., Zhang, J., Humaidi, A. J., Al-Dujaili, A., Duan, Y., Al-Shamma, O., Santamaría, J., Fadhel, M. A., Al-Amidie, M., & Farhan, L. (2021). Review

of deep learning: concepts, CNN architectures, challenges, applications, future directions. *Journal of Big Data*, *8*(1). https://doi.org/10.1186/s40537-021-00444-8

Andika Surya, I. M., Cahyanto, T. A., & Muharom, L. A. (2025). Deep Learning dengan Teknik Early Stopping untuk Mendeteksi Malware pada Perangkat IoT. *Jurnal Teknologi Informasi Dan Ilmu Komputer*, *12*(1), 21–30. https://doi.org/10.25126/jtiik.2025128267

Andrew G. Howard, M. Z. B. C. D. K. W. W. T. W. M. A. H. A. (2017). MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications. *Arxiv*. https://doi.org/10.48550/arXiv.1704.04861

Ashraf, M., Abid, F., Din, I. U., Rasheed, J., Yesiltepe, M., Yeo, S. F., & Ersoy, M. T. (2023). A Hybrid CNN and RNN Variant Model for Music Classification. *Mdpi*, *13*(3). https://doi.org/10.3390/app13031476

Asrafil, A., Paliwang, A., Ridwan, M., Septian, D., Cahyanti, M., Ericks, D., Swedia, R., & Informatika, J. T. (2020). KLASIFIKASI PENYAKIT TANAMAN APEL DARI CITRA DAUN DENGAN CONVOLUTIONAL NEURAL NETWORK. *Sebatik*. https://doi.org/10.46984/sebatik.v24i2.1060

Dutta, J., & Chanda, D. (2024). MUSIC EMOTION RECOGNITION AND CLASSIFICATION USING HYBRID CNN-LSTM DEEP NEURAL NETWORK. *Bangladesh Journal of Multidisciplinary Scientific Research*, *9*(3), 21–32. https://doi.org/10.46281/bjmsr.v9i3.2230

Falola, P., Alabi, E., Folashade, O., & Fasae, O. D. (2022). MUSIC GENRE CLASSIFICATION USING MACHINE AND DEEP LEARNING TECHNIQUES: A REVIEW. *Reserchjet*. https://doi.org/10.17605/OSF.IO/FZQXW

Fardhani, S. M., Wihardi, Y., & Piantari, E. (2021). *Klasifikasi Genre Musik Dengan Mel Frequency Cepstral Coefficient Dan Spektogram Menggunakan Convolutional Neural Network* (Vol. 4, Issue 1). https://doi.org/https://doi.org/10.17509/jatikom.v4i1.41465

Khoirun Nisa', N., & Riadi, A. A. (2025). Klasifikasi Wayang Kulit Kurawa Menggunakan Algoritma CNN Classification of Wayang Kulit Kurawa Using CNN Algorithm. *Jurnal Pendidikan Dan Teknologi Indonesia (JPTI)*, *5*(6), 1799–1808. https://doi.org/10.52436/1.jpti.856

Li, T. (2024). Optimizing the configuration of deep learning models for music genre classification. *Heliyon*, *10*(2). https://doi.org/10.1016/j.heliyon.2024.e24892

Purnama, N. (2022). Music Genre Recommendations Based on Spectrogram Analysis Using Convolutional Neural Network Algorithm with RESNET-50 and VGG-16 Architecture. *JISA*. https://trilogi.ac.id/journal/ks/index.php/JISA/article/view/1270

Reza Fahcruroji, A., Yunita Wijaya, M., Fauziah, I., Sains dan Teknologi, F., Syarif Hidayatullah Jakarta Jl Ir Juanda No, U. H., Ciputat Tim, K., & Tangerang Selatan, K. (2024). IMPLEMENTASI ALGORITMA CNN MOBILENET UNTUK KLASIFIKASI GAMBAR SAMPAH DI BANK SAMPAH. *Prosisko*. https://doi.org/10.30656/prosisko.v11i1.8101

Sridhar, A. (2024). Attention-guided Spectrogram Sequence Modeling with CNNs for Music Genre Classification. *Arxiv*. http://arxiv.org/abs/2411.14474

Tzanetakis, G., & Cook, P. (2002). Musical genre classification of audio signals. *IEEE Transactions on Speech and Audio Processing*, *10*(5), 293–302. https://doi.org/10.1109/TSA.2002.800560

Wairata, C. R., Swedia, E. R., & Cahyanti, M. (2021). PENGKLASIFIKASIAN GENRE MUSIK INDONESIA MENGGUNAKAN CONVOLUTIONAL NEURAL NETWORK. *Sebatik*, *25*(1), 255–261. https://doi.org/10.46984/sebatik.v25i1.1286

Yehezkiel, S. Y., & Suyanto, Y. (2022). Music Genre Identification Using SVM and MFCC Feature Extraction. *IJEIS (Indonesian Journal of Electronics and Instrumentation Systems)*, *12*(2), 115. https://doi.org/10.22146/ijeis.70898