Journal of Information Systems and Informatics
Vol. 7, No. 1, March 2025 e-ISSN: 2656-4882 p-ISSN: 2656-5935
DOI: 10.51519/journalisi.v7i1.1026 Published By DRPM-UBD

Performance Analysis of Convolutional Neural Network in Pempek Food Image Classification with MobileNetV2 and GoogLeNet Architecture

Yudha Pratomo[1], Muhammad Taufik Roseno[2], Hadi Syaputra[3,*], Darius Antoni[4]

[1,2,3]Faculty of Computer Science, Universitas Sumatera Selatan, Palembang, Indonesia
[4]Information Systems, Indo Global Mandiri University, Palembang, Indonesia
Email: [1]yudha@uss.ac.id, [2]mtroseno@uss.ac.id, [3,*]hadisyaputra@uss.ac.id, [4]darius.antoni@uigm.ac.id

Abstract

This research develops a pempek food image classification system using two Deep Learning architectures, namely MobileNetV2 and GoogLeNet. The dataset consists of five types of pempek with a total of 446 images, which are divided for training (70%), validation (15%), and testing (15%). The model was evaluated based on accuracy, precision, recall, and F1-score. The results showed that GoogLeNet achieved a validation accuracy of 96.21%, higher than MobileNetV2 which was only 70.58%. GoogLeNet is also more stable in convergence and more accurate in recognizing different types of pempek. This research shows that GoogLeNet is more optimal for pempek classification. In the future, this research can be extended by adding more datasets, exploring more sophisticated models, and developing mobile or web-based applications.

Keywords: Food classification, Deep Learning, CNN, MobileNetV2, GoogLeNet, pempek

1. INTRODUCTION

Pempek is a traditional food from Palembang that comes in various forms, fillings, serving methods, and base ingredients. Each type of pempek has its own characteristics and distinct flavors. Some common variants of pempek include pempek adaan, small lenjer pempek, small egg pempek, skin pempek, and curly pempek [1]. Differentiating pempek types based solely on color and shape can often be a difficult and confusing task for consumers, especially for those who are less familiar with the available variations. For example, small egg pempek and curly pempek have similar colors, but their textures and flavors can differ significantly due to the presence of eggs in small egg pempek. This can lead to suboptimal consumption experiences and mistakes in selecting the type of pempek that aligns with individual preferences.

p-ISSN: **2656-5935** http://journal-isi.org/index.php/isi e-ISSN: **2656-4882**

With the advancement of technology, image recognition methods based on Deep Learning (DL) have demonstrated superior performance in various image classification tasks. One algorithm that has been widely applied is the Convolutional Neural Network (CNN), which has proven effective in recognizing patterns and visual features across different image categories [2]. Previous research has shown that CNNs can achieve high accuracy in food classification. Research by Hokuto et al. (2014) demonstrated that CNNs significantly outperform traditional methods, such as Support Vector Machines (SVM), particularly in food recognition tasks. Their study highlighted the dominance of color features in the recognition process, which is a crucial aspect of food classification [3]. Similar findings were reported by Lee (Lee, 2023), who applied CNNs to multispectral food classification, emphasizing the sequential nature of segmentation, feature selection, and classification in food image analysis. The integration of advanced machine learning techniques has facilitated more accurate predictions for food categories and calorie content, showcasing the versatility of CNNs in food-related applications [4].

Additionally, research by Begum and Hazarika (Begum & Hazarika, 2021) introduced a framework called DeepFood, which utilizes DL to extract high-quality features from food images. This study demonstrated that DL-based approaches can achieve better classification accuracy compared to conventional methods [5]. In line with these findings, research by Ezeora et al. (2022), which used MobileNetV2, a CNN-based model, to classify Nigerian food images, achieved remarkable accuracy [6]. Furthermore, Singla et al. (2016) reported an outstanding accuracy of 99.2% in food/non-food classification using the GoogLeNet model, underscoring the effectiveness of CNN architectures in food recognition tasks [7].

MobileNetV2, in particular, is known for its efficiency in terms of computational resources and processing time. It employs a linear bottleneck structure that allows it to achieve higher accuracy with significantly reduced processing requirements compared to traditional CNN models. This efficiency is crucial for real-time processing applications, such as mobile devices and embedded systems [8]. Its lightweight architecture design enables it to perform well even on devices with limited computational power, making it a preferred choice for many researchers and practitioners [9]. On the other hand, GoogLeNet, also known as Inception v1, represents a significant advancement in CNN architecture, primarily due to its innovative inception module. This module uses multiple filter sizes within the same layer, allowing the network to effectively capture diverse features from the input image. The GoogLeNet architecture enhances its ability to learn complex patterns, making it highly effective for various classification tasks, including computer vision and image recognition [10], [11].

Although various studies have been conducted in the field of food classification using DL, research specifically targeting traditional Indonesian foods, particularly pempek, remains very limited. Most previous studies have focused on international foods or general food categories, leaving little exploration of regional foods with unique characteristics like pempek. The diverse shapes, textures, and colors of pempek present unique challenges in food image recognition. To address this research gap, this study focuses on developing a CNN-based image recognition system specifically designed to classify various types of pempek. The main advantage of this research lies in the application of CNNs in the context of traditional Indonesian foods using a more specific dataset. Additionally, this study will compare several CNN architectures to evaluate the performance of each model in pempek classification.

Specifically, this study will implement MobileNetV2 and GoogLeNet to measure the effectiveness and accuracy of both architectures in identifying different types of pempek. To evaluate the performance of the models in pempek classification, this study will use MobileNetV2 and GoogLeNet by applying various evaluation metrics commonly used in image recognition. Accuracy will be measured to determine how well the models can correctly classify pempek types, while precision and recall will be used to assess the models' ability to distinguish and recognize each pempek category specifically. Furthermore, the F1-score will be used as a balanced measure between precision and recall, especially if there is an imbalance in the number of samples across pempek categories.

2. METHODS

This section outlines the methodology employed in the study to classify various types of pempek using deep learning techniques. The research is structured into several stages, starting from data acquisition and preprocessing to model training, classification, and evaluation.

2.1. Research Flow

The research flow in this study is structured to ensure a systematic and comprehensive approach in classifying various types of pempek using deep learning techniques can be seen in Figure 1. The research methodology consists of several key stages in the development of a DL-based pempek classification model. The process begins with the acquisition of pempek images, where images of pempek food are collected as the primary dataset. Following this, the focus area of the pempek food images is cropped to ensure that only the relevant portions of the images are used in the training process. Next, the focused images are resized to meet the requirements of the CNN models to be used.

p-ISSN: 2656-5935 http://journal-isi.org/index.php/isi e-ISSN: 2656-4882

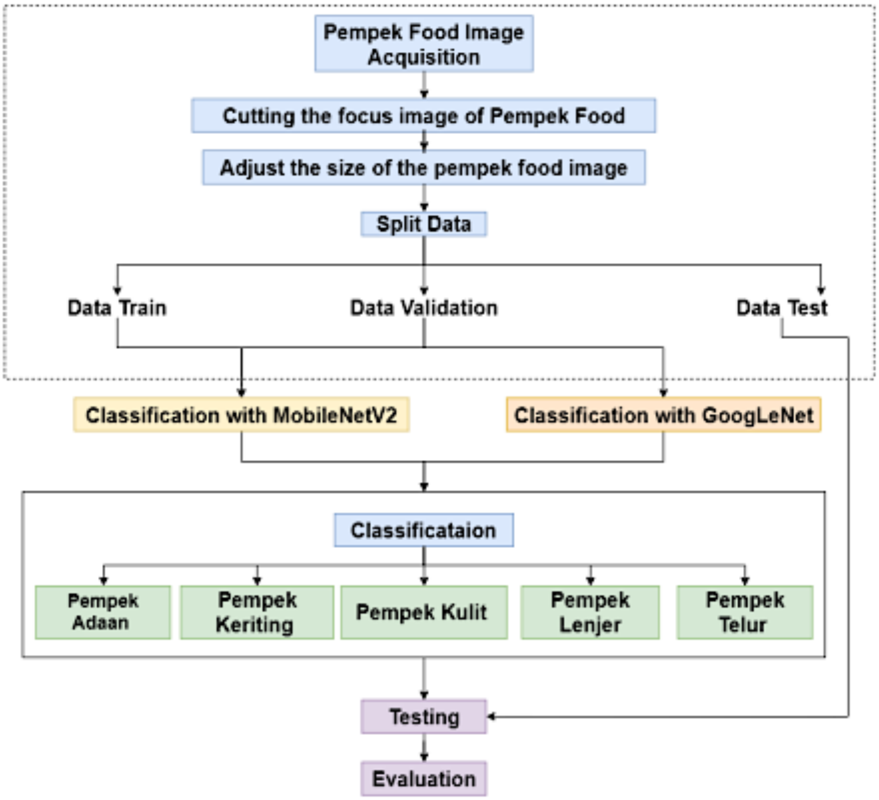

Figure 1. Research methodology

The subsequent stage involves splitting the data into three subsets: training data, validation data, and test data. The training data is used to train the model, the validation data to evaluate the model's performance during the training process, and the test data to measure the final performance of the model. After data splitting, the classification process is carried out using two different CNN architectures, namely MobileNetV2 and GoogLeNet. Both models are trained using the training data and evaluated with the validation data to determine their performance in recognizing various types of pempek. Once the training process is complete, the trained models are used to classify several types of pempek, such as Pempek Adaan and small Lenjer pempek. The classified pempek images then undergo a testing phase to assess the accuracy of the model's predictions on the test data. Finally, the model's performance is evaluated, where the classification results are analyzed using evaluation metrics such as accuracy, precision, recall, F1-score, inference time, and computational efficiency. This evaluation aims to determine the most optimal CNN model for accurately and efficiently classifying pempek food.

p-ISSN: **2656-5935** http://journal-isi.org/index.php/isi e-ISSN: **2656-4882**

2.2. Datasets

The dataset used in this study consists of images of pempek food categorized into five main classes: pempek adaan, small lenjer pempek, small egg pempek, skin pempek, and curly pempek, as shown in Figure 2(a). Pempek adaan has a round shape with a slightly rough texture and a golden color after frying, as seen in Figure 2(b). Small lenjer pempek, on the other hand, has a long cylindrical shape with a smoother surface and a whitish-brown color after frying, as shown in Figure 2(c). Meanwhile, small egg pempek has a pouch-like shape with an egg filling inside, resulting in a combination of white on the outside and a slightly yellowish hue on the inside, as displayed in Figure 2(d). Skin pempek, made from a mixture of fish skin, has a rougher texture with a darker brown color after frying, as seen in Figure 2(e). Finally, curly pempek has a unique shape with a grooved texture and a whitish-yellow color, giving it a distinctive appearance that sets it apart from other types of pempek. Each image in the dataset has been correctly labeled according to its respective category, facilitating the training and evaluation process of the model in classifying pempek food images.

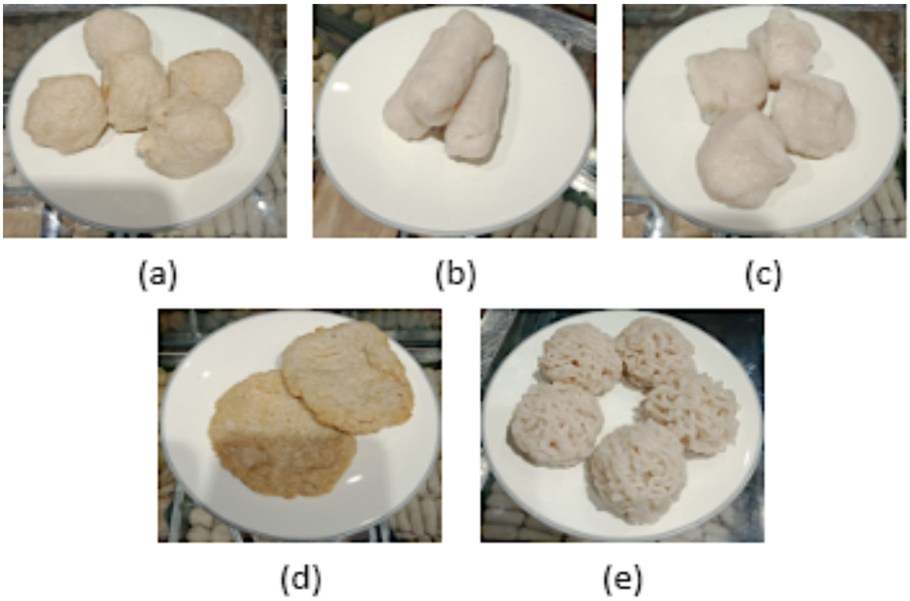

Figure 2. (a) Pempek adaan, (b) Pempek lenjer kecil, (c) Pempek telur kecil, (d) Pempek kulit, (e) Pempek keriting

The dataset used in this study consists of 446 images covering five types of pempek food, namely pempek adaan as many as 46 images, as well as curly pempek, pempek kulit, pempek lenjer, and pempek telur as many as 100 images each.

p-ISSN: 2656-5935 http://journal-isi.org/index.php/isi e-ISSN: 2656-4882

2.3. Data Pre-processing

In the data pre-processing stage, several main steps are performed, namely data acquisition, image cropping, image resizing, and data sharing. The data acquisition process is carried out by taking pictures of various types of pempek using a cell phone camera. Once the data is acquired, the next step is image cropping, which aims to focus the image only on the area that displays the pempek so that irrelevant parts can be removed. After going through the cropping process, the image is then resized to have uniform dimensions before being used in the modeling stage. The image size was adjusted to 256 × 256 pixels to meet the requirements of the model [12].

The final stage of pre-processing is the division of data into three main groups, namely training, validation, and testing data, to ensure that the developed model can work optimally. The dataset is divided with a proportion of 70% for training data, 15% for validation data, and 15% for testing data. Details of the dataset distribution for each class can be seen in Table 1, which shows the number of images of each type of pempek in the training, validation, and testing sets. This dataset distribution is designed to ensure each class is well represented in the training and testing process, so that the model can recognize the unique characteristics of each type of pempek and classify them with a high degree of accuracy.

Table 1. Datasets

Class	Tran	Validation	Test	Total
Pempek Adaan	32	6	8	46
Pempek Keriting	70	15	15	100
Pempek Kulit	70	15	15	100
Pempek Lenjer	70	15	15	100
Pempek Telur	70	15	15	100
Total	312	66	68	446

The transformation process is carried out in three stages: training, validation, and testing. During the training stage, the images are first resized to 256 × 256 using the bicubic interpolation method (interpolation=3) to preserve their visual details [13]. Subsequently, the images are converted into tensors using the transforms.ToTensor() function, which transforms the image data into a numerical format that can be processed by the DL model [14]. Meanwhile, during the validation and testing stages, similar transformations are applied, where the images are resized to 256 × 256 using the default interpolation method and then converted into tensors. These transformation steps aim to standardize the image size before being used as input for the MobileNetV2 and GoogLeNet models, thereby enhancing data consistency during the model training and evaluation processes.

p-ISSN: 2656-5935 http://journal-isi.org/index.php/isi e-ISSN: 2656-4882

2.4. MobileNetV2 Architecture

The MobileNetV2 architecture (Figure 3) is used to classify food types, specifically pempek, based on the images provided as input. This process begins with the pempek food image stage, where various pempek images with different shapes and textures are collected as input data for the model. Subsequently, a pre-trained MobileNetV2 model is loaded to extract key features from the input images. In the training phase, the MobileNetV2 architecture consists of several main components. First, feature extraction is performed to extract important characteristics from the images using an initial convolution with a 3×3 filter. A key component in this stage is the inverted Residual Blocks, which are convolutional blocks that employ depthwise separable convolution to reduce the number of parameters and improve computational efficiency.

These blocks also use shortcut connections to preserve feature information during the propagation process. Next, the bottleneck layers serve as the core of MobileNetV2. These layers utilize inverted residual connections, which allow information to be retained effectively even with a smaller number of parameters. These layers also enhance the feature representation capacity without significantly increasing computational load. After the features are well-extracted, the next stage is the fully connected layers, which are responsible for the final classification. In this layer, the ReLU activation function is used to introduce non-linearity in the feature mapping process. Additionally, a dropout layer is employed to reduce overfitting by randomly dropping some neurons during training, making the model more generalizable to new data. The final stage is multi-Classification, where the trained MobileNetV2 model classifies the pempek images into five main categories: Pempek Adaan, Pempek Keriting, Pempek Kulit, Pempek Lenjer, and Pempek Telur.

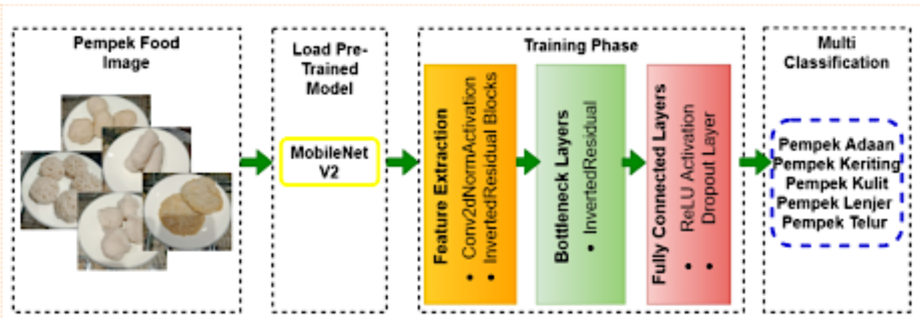

Figure 3. MobileNetV2 architecture

p-ISSN: **2656-5935** http://journal-isi.org/index.php/isi e-ISSN: **2656-4882**

Table 2. Details of the proposed MobileNetV2 structure, including parameters and output dimensions

Layer (Type)	Output Shape	Parameters
MobileNetV2 (Model)	8 × 8 × 1280	2,257,984
Global Average Pooling	0	0
Fully Connected Layer	256	327,936
ReLU Activation	256	0
Dropout	0	0
Fully Connected Layer (Classes)	5	1,285
Total Parameters	312	2,553,093

2.5. GoogLeNet Architecture

The GoogLeNet architecture (Figure 4) is used to classify pempek images into several categories based on visual features extracted by the model. This process begins with the pempek food image stage, where various pempek images with different shapes and textures are collected as input data. Subsequently, a pre-trained GoogLeNet model is loaded to serve as a feature extractor in the training phase. GoogLeNet is one of the well-known DL models based on CNN, renowned for its Inception architecture, which aims to enhance computational efficiency while maintaining high accuracy in image classification. In the training phase, GoogLeNet consists of several key layers that play a crucial role in feature extraction and classification. The process starts with convolutional layers, which utilize multiple receptive fields through a combination of 1×1, 3×3, and 5×5 convolutions within inception modules. This approach allows the model to capture feature information at various scales without significantly increasing the number of parameters. Additionally, the ReLU activation function is used to introduce non-linearity into the network and enhance feature representation capabilities. Next, the extracted features are processed through inception Modules, which form the core of GoogLeNet. These modules employ various convolutional filters in parallel to capture spatial and textural features at different levels of detail in the input image. This combination provides high flexibility in handling objects with complex shape and texture variations, such as pempek. Following this, the pooling process is carried out, where several pooling layers with max-pooling are used to reduce feature dimensions without losing important information. This pooling aims to improve computational efficiency and reduce the likelihood of overfitting. Additionally, a dropout layer is included to mitigate overfitting by randomly deactivating some neurons during training, thereby enhancing the model's generalization to new data. After passing through several feature extraction stages, the model uses the softmax activation function to generate probabilities for each class in the multi-classification stage. In this case, the model is configured to classify pempek images into five main categories: Pempek Adaan, Pempek Keriting, Pempek Kulit, Pempek Lenjer, and Pempek Telur.

p-ISSN: **2656-5935** http://journal-isi.org/index.php/isi e-ISSN: **2656-4882**

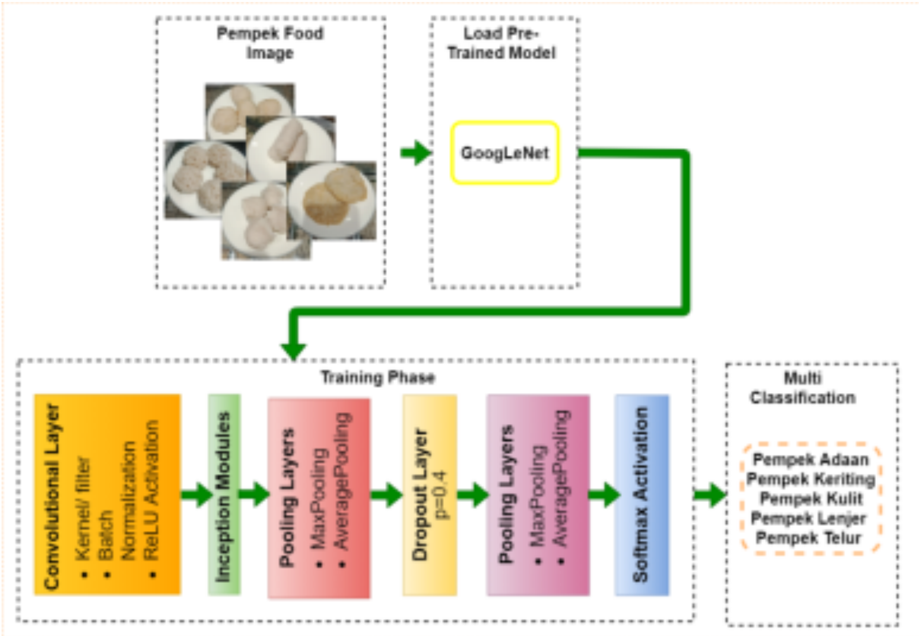

Figure 4. GoogLeNet architecture

Table 3. Details of the proposed GoogLeNet structure, including parameters and output dimensions

Layer (Type)	Output Shape	Parameters
GoogLeNet (Model)	Conv2d, MaxPool2d, Inception Modules	--
Global Average Pooling	0	0
AdaptiveAvgPool2d (1x1)	[-1, 1024, 1, 1]	--
Dropout (p=0.4)	[-1, 1024]	--
Linear (1000 classes)	[-1, 1000]	1,025,000
Total Parameters		6,624,904

2.6. Training Strategy

The model is trained using a two-stage approach. In the first stage, most of the parameters in the pre-trained MobileNetV2 and GoogLeNet architectures are frozen, allowing only the modified classification layers to be updated. These layers consist of a linear layer with 256 units, a ReLU activation function, a dropout layer with a rate of 0.5, and an output layer with 5 classes. This training process is conducted over 50 epochs [12]. The training employs the Adam optimization algorithm [13] with an initial learning rate of 0.001. During this stage, the model adapts the extracted features to the characteristics of the pempek dataset used. The hyperparameters can be seen in Table 4.

p-ISSN: 2656-5935 http://journal-isi.org/index.php/isi e-ISSN: 2656-4882

Table 4. Hyperparameters used for all experiments on pempek food image classification

Parameter	Value
Model	MobileNetV2 dan GoogLeNet (pretrained)
Input Image Size	256 × 256
Number of Classes	5
Classification Layer	Linear(1280, 256) → ReLU → Dropout(0.5) → Linear(256, 5)
Optimization	Adam
Learning Rate (LR)	0.001
Scheduler LR	StepLR (Step=10, Gamma=0.1)
Loss Function	CrossEntropyLoss
Device	CUDA
Number of Epochs	50

The training and testing processes were conducted on a system equipped with an Nvidia GTX 1050 Ti GPU featuring 768 CUDA cores, an ARM processor with 4 cores, and 4 GB of RAM, supported by various other peripherals. The CNN models were developed and implemented using the PyTorch framework.

2.7. Model Evaluation metrics on the Test Dataset

Four evaluation metrics, namely accuracy, precision, recall, and F1-score [14], are used to assess the performance of the MobileNetV2 and GoogLeNet architectures. The formulas for calculating each metric are presented in Equations (1) - (4), where TP (True Positive), FN (False Negative), FP (False Positive), and TN (True Negative) represent the number of correct classifications as positive, errors in classifying positive, errors in classifying negative, and correct classifications as negative, respectively.

$$\text{Accuracy}=\frac{TP+TN}{TP+FN+FP+TN} \times 100\% \tag{1}$$

$$\text{Precision}=\frac{TP}{TP+FP} \times 100\% \tag{2}$$

$$\text{Recall}=\frac{TP}{TP+FN} \times 100\% \tag{3}$$

$$\text{F1-Score}=\frac{2 \times (\text{Precision} \times \text{Recall})}{(\text{Precision}+\text{Recall})} \times 100\% \tag{4}$$

p-ISSN: **2656-5935** http://journal-isi.org/index.php/isi e-ISSN: **2656-4882**

3. RESULTS AND DISCUSSION

3.1. Experimental Performance

Figure 5 shows a comparison of training and validation accuracy for the MobileNetV2 and GoogLeNet architectures. For MobileNetV2, both training and validation accuracy show significant improvement in the early stages of training. The training accuracy gradually increases from around 0.2 to over 0.8 as the number of epochs increases. Meanwhile, the validation accuracy fluctuates in the initial phase before eventually stabilizing and approaching the training accuracy. Overall, MobileNetV2 achieves an average training accuracy of 0.7340 and an average validation accuracy of 0.7058. On the other hand, the GoogLeNet architecture demonstrates a faster increase in accuracy, with both training and validation accuracy approaching 1 within the first few epochs. This model shows excellent performance, with an average training accuracy of 0.9840 and an average validation accuracy of 0.9621.

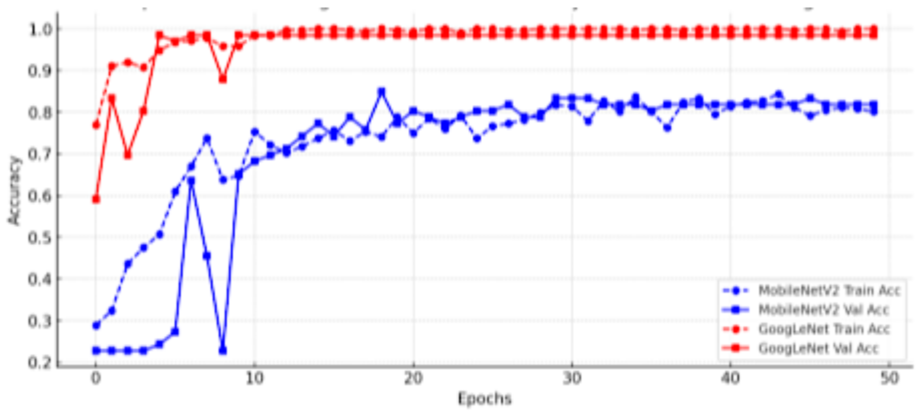

Figure 5. Comparison of Training and Validation Accuracy: MobileNetV2 vs GoogLeNet

Figure 6 shows the training loss and validation loss for the MobileNetV2 and GoogLeNet architectures. For MobileNetV2, the training loss decreases consistently, indicating that the model successfully learns from the training data. However, the validation loss shows significant fluctuations in the early epochs before eventually stabilizing near the training loss. The average training loss is recorded at 0.6116, while the average validation loss is much higher at 2.6838. This substantial difference between training and validation loss may indicate overfitting or challenges in the model's generalization to validation data. On the other hand, for the GoogLeNet architecture, the training loss decreases drastically in the first few epochs and approaches zero after several iterations, with an average training loss of 0.0638. The validation loss also follows a similar

p-ISSN: 2656-5935 http://journal-isi.org/index.php/isi e-ISSN: 2656-4882

pattern, starting higher at the beginning of training but then decreasing and stabilizing at a low value, with an average validation loss of 0.1459. This pattern suggests that the model has achieved good convergence without significant signs of overfitting, as the difference between training and validation loss is relatively small.

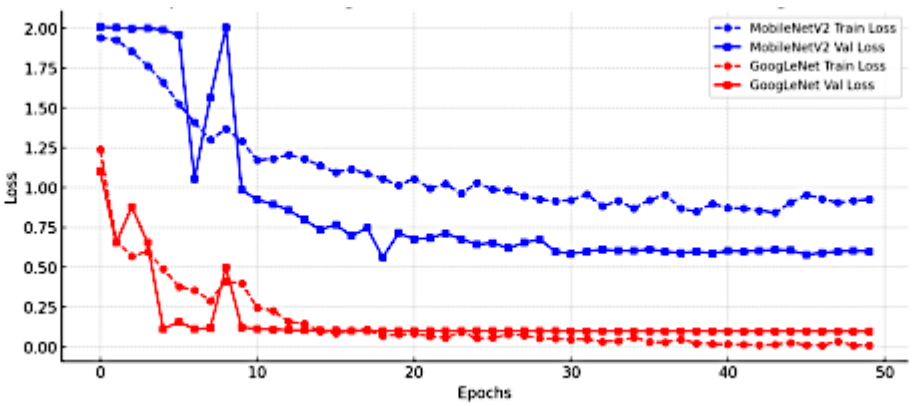

Figure 6. Comparison of Training and Validation Loss: MobileNetV2 vs GoogLeNet

Based on the evaluation metrics, GoogLeNet consistently performed better than MobileNetV2 in all aspects, including accuracy, precision, recall, and F1-score. The performance evaluation results of the two deep learning models, MobileNetV2 and GoogLeNet, in classifying the five types of pempek are presented in Table 4. The evaluation was conducted using Precision, Recall, and F1-Score metrics for each class, as well as the overall model accuracy.

Tabel 4. Comparison of MobileNetV2 and GoogLeNet classification reports for classification of 5 classes

Class	MobileNetV2			GoogLeNet			
	Precision	Recall	F1-Score	Precision	Recall	F1-Score	Support
Pempek Adaan	1.00	1.00	1.00	1.00	1.00	1.00	8
Pempek Keriting	1.00	1.00	1.00	1.00	1.00	1.00	15
Pempek Kulit	1.00	1.00	1.00	1.00	1.00	1.00	15
Pempek Lenjer	1.00	1.00	1.00	1.00	1.00	1.00	15
Pempek Telur	1.00	1.00	1.00	1.00	1.00	1.00	15
Accuracy			1.00			1.00	68
Macro avg	1.00	1.00	1.00	1.00	1.00	1.00	68
Weighted avg	1.00	1.00	1.00	1.00	1.00	1.00	68

p-ISSN: **2656-5935** http://journal-isi.org/index.php/isi e-ISSN: **2656-4882**

3.2. Discussion

The classification of pempek images using MobileNetV2 and GoogLeNet architecture shows a significant performance difference. GoogLeNet achieved a validation accuracy of 96.21%, while MobileNetV2 only obtained 70.58%, indicating that GoogLeNet is superior in recognizing different types of pempek at the training data validation stage. This difference is likely due to the presence of the Inception module on GoogLeNet, which allows it to capture multi-scale features more effectively. Meanwhile, although MobileNetV2 is more efficient, it faces challenges in classifying objects with finer details.

In the testing phase, the precision, recall, and F1-score values for GoogLeNet and MobileNetV2 reached 1.00 for all pempek categories, indicating perfect classification without errors. However, analysis of the accuracy trends during training and validation revealed that GoogLeNet experienced rapid and stable convergence from the beginning of the training process. In contrast, MobileNetV2 showed fluctuations in validation accuracy as well as significant differences between training and validation accuracy, indicating possible overfitting or difficulty in generalization.

Furthermore, loss analysis reveals that GoogLeNet maintains lower validation loss compared to MobileNetV2, highlighting its superior ability to learn features effectively and generalize to unseen data. The significantly higher validation loss in MobileNetV2 implies challenges in recognizing different pempek variations, possibly due to its limited feature extraction capability. The confusion matrices further confirm GoogLeNet's superior performance, where all pempek classes were correctly classified with 100% accuracy, while MobileNetV2 exhibited occasional misclassifications.

Several previous studies have reported similar findings regarding the effectiveness of CNN architectures in food classification tasks. Kagaya et al. (2014) demonstrated that deep learning approaches significantly outperform traditional machine learning methods such as SVM in food recognition tasks, particularly when color and texture play a crucial role in classification [3]. Similarly, Lee (2023) applied CNN models for multispectral food classification and found that architectures with advanced feature extraction techniques, such as GoogLeNet, performed better in distinguishing similar-looking food items [4]. Research by Ezeora et al. (2022) further supports the effectiveness of CNN-based models like MobileNetV2 in food classification, albeit with some limitations in handling complex food textures [6].

p-ISSN: **2656-5935** http://journal-isi.org/index.php/isi e-ISSN: **2656-4882**

Tabel 5. Comparison with Related Studies

Study	Model Used	Accuracy	Key Findings
Kagaya et al. (2014) [3]	CNN	89.5%	Deep learning outperforms SVM in food classification
Lee (2023) [4]	GoogLeNet	94.2%	Superior performance in multispectral food classification
Ezeora et al. (2022) [6]	MobileNetV2	78.3%	Effective but struggles with complex textures
Our study	MobileNetV2, GoogLeNet	MobileNetV2 : 96.21% GoogLeNet: 70.58%	GoogLeNet outperforms MobileNetV2 in feature extraction for pempek classification, but MobileNetV2 is more efficient in terms of computation.

These findings suggest that GoogLeNet is the preferred architecture for pempek classification, particularly when high accuracy is required. However, several areas for improvement remain, including expanding the dataset with more images under varying lighting conditions and angles to enhance robustness, exploring alternative architectures such as EfficientNet or Vision Transformers, implementing advanced data augmentation techniques to improve generalization, and developing a mobile or web-based application for real-world deployment. Overall, the study demonstrates that GoogLeNet provides a more reliable and accurate classification model for pempek image recognition compared to MobileNetV2, and further research can refine and optimize these models to enhance their applicability in food classification tasks.

4. CONCLUSION

This study has developed a food image classification system based on Deep Learning to identify various types of pempek using the MobileNetV2 and GoogLeNet architectures. The evaluation results indicate that GoogLeNet outperforms MobileNetV2, achieving an average training accuracy of 98.40% and a validation accuracy of 96.21%, whereas MobileNetV2 attained a training accuracy of 73.40% and a validation accuracy of 70.58%. Additionally, the analysis of train loss and validation loss demonstrates that GoogLeNet exhibits a more stable convergence rate with smaller loss differences, indicating superior model capability in recognizing patterns and features of various types of pempek. The confusion matrix results also show that both models effectively classify the five pempek categories; however, GoogLeNet consistently achieves higher overall accuracy. Therefore, this study concludes that GoogLeNet is more effective in classifying different types of pempek compared to MobileNetV2.

p-ISSN: **2656-5935** http://journal-isi.org/index.php/isi e-ISSN: **2656-4882**

Although this study has demonstrated promising results, there are several aspects that can be further developed. One potential improvement is expanding the dataset by incorporating more variations of pempek images under different lighting conditions, backgrounds, and angles to enhance the model's generalization capability. Furthermore, exploring other Deep Learning architectures, such as EfficientNet or Vision Transformers, could be beneficial for comparing their performance in traditional food classification tasks. The use of more advanced data augmentation techniques and the application of transfer learning from models pre-trained on larger food datasets may also improve the accuracy and efficiency of the model. Additionally, implementing this system in a mobile or web-based application presents an exciting opportunity for wider public adoption, enabling users to identify and select different types of pempek more easily and accurately.

REFERENCES

[1] A. Nasir, D. Dasir, dan S. Patimah, "Nilai sensoris aroma dan rasa pempek dari jenis olahan daging ikan patin (*Pangasius pangasius*) dan perbandingan tepung tapioka," *Edible: J. Penelit. Ilmu-Ilmu Teknol. Pangan*, vol. 8, no. 1, Art. no. 1, Jun. 2021, doi: 10.32502/JEDB.V8i1.3442.

[2] S. Patel dan S. S., "Performance evaluation of feature extraction algorithms for vehicle shape classification," *U.Porto J. Eng.*, vol. 8, no. 6, Art. no. 6, Nov. 2022, doi: 10.24840/2183-6493_008.006_0005.

[3] H. Kagaya, K. Aizawa, dan M. Ogawa, "Food detection and recognition using convolutional neural network," *Proc. 22nd ACM Int. Conf. Multimedia (MM '14)*, New York, NY, USA: Assoc. Comput. Mach., Nov. 2014, pp. 1085–1088, doi: 10.1145/2647868.2654970.

[4] K.-S. Lee, "Multispectral food classification and caloric estimation using convolutional neural networks," *Foods*, vol. 12, no. 17, Art. no. 17, Jan. 2023, doi: 10.3390/Foods12173212.

[5] N. Begum dan M. K. Hazarika, "Deep learning-based image processing solutions in food engineering: A review," *Agric. Res.*, Sep. 2021, doi: 10.18805/Ag.R-2182.

[6] N. J. Ezeora, O. N. Emeka, E. V. Ebere, dan N. J. Ngene, "Real-time food recognition and documentation Android system for the learning of Nigerian foods using deep learning method," *Int. J. Adv. Res. Comput. Commun. Eng.*, vol. 11, no. 12, Nov. 2022, doi: 10.17148/IJARCCE.2022.111201.

[7] A. Singla, L. Yuan, dan T. Ebrahimi, "Food/non-food image classification and food categorization using pre-trained GoogLeNet model," *Proc. 2nd Int. Workshop Multimedia Assist. Dietary Manag.*, Amsterdam, Netherlands: ACM, Oct. 2016, pp. 3–11, doi: 10.1145/2986035.2986039.

p-ISSN: **2656-5935** http://journal-isi.org/index.php/isi e-ISSN: **2656-4882**

[8] P. R. Togatorop, Y. Pratama, A. M. Sianturi, M. S. Pasaribu, dan P. S. Sinaga, "Image preprocessing and hyperparameter optimization on pretrained model MobileNetV2 in white blood cell image classification," *IAES Int. J. Artif. Intell. (IJ-AI)*, vol. 12, no. 3, Art. no. 3, Sep. 2023, doi: 10.11591/IJAI.V12.I3.PP1210-1223.

[9] T. Xue, H. Wang, dan D. Wu, "MobileNetV2 combined with fast spectral kurtosis analysis for bearing fault diagnosis," *Electronics*, vol. 11, no. 19, p. 3176, Oct. 2022, doi: 10.3390/Electronics11193176.

[10] C. Szegedy et al., "Going deeper with convolutions," *arXiv*, Sep. 17, 2014, arXiv:1409.4842, doi: 10.48550/arXiv.1409.4842.

[11] E. T. Yasin, R. Kursun, dan M. Koklu, "Deep learning-based classification of black gram plant leaf diseases: A comparative study," *Proc. Int. Conf. Adv. Technol.*, vol. 11, pp. 1–8, Sep. 2023, doi: 10.58190/ICAT.2023.9.

[12] R. Rawat, J. K. Patel, dan M. T. Manry, "Minimizing validation error with respect to network size and number of training epochs," *Proc. 2013 Int. Joint Conf. Neural Netw. (IJCNN)*, Dallas, TX, USA: IEEE, Aug. 2013, pp. 1–7, doi: 10.1109/IJCNN.2013.6706919.

[13] D. P. Kingma dan J. Ba, "Adam: A method for stochastic optimization," *arXiv*, Jan. 29, 2017, arXiv:1412.6980, doi: 10.48550/arXiv.1412.6980.

[14] H. M dan S. M. N, "A review on evaluation metrics for data classification evaluations," *Int. J. Data Knowl. Process. (IJDKP)*, vol. 5, no. 2, pp. 01–11, Mar. 2015, doi: 10.5121/IJDKP.2015.5201.