International Journal Software Engineering and Computer Science (IJSECS)
5 (1), 2025, 153-163
Published Online April 2025 in IJSECS (http://www.journal.lembagakita.org/index.php/ijsecs)
P-ISSN: 2776-4869, E-ISSN: 2776-3242. DOI: https://doi.org/10.35870/ijsecs.v5i1.3376.

RESEARCH ARTICLE **Open Access**

Deep Learning for Classification of Mammalian Reproduction

Nur Mufidah *
Master of Applied Computer Engineering, Politeknik Caltex Riau, Pekanbaru City, Riau Province, Indonesia.
Corresponding Email: nur23mttk@mahasiswa.pcr.ac.id.

Yuliska Zakaria
Master of Applied Computer Engineering, Politeknik Caltex Riau, Pekanbaru City, Riau Province, Indonesia.
Email: yuliska@pcr.ac.id.

Mutia Sari Zulvi
Master of Applied Computer Engineering, Politeknik Caltex Riau, Pekanbaru City, Riau Province, Indonesia.
Email: mutia@pcr.ac.id.

Received: November 25, 2024; Accepted: January 20, 2025; Published: April 1, 2025.

Abstract: This research aims to classify mammalian reproduction using deep learning techniques, specifically focusing on Convolutional Neural Networks (CNN), VGG16, and MobileNetV2. CNN is applied to extract visual features from images related to mammalian reproductive systems, while the VGG16 and MobileNetV2 architectures are utilized to enhance accuracy and efficiency in classification. The dataset used consists of images of mammalian reproductive organs, analyzed using data augmentation techniques to improve model reliability. Data augmentation includes various transformations such as rotation, zoom, flipping, and brightness adjustments to enrich the dataset's variety and reduce the risk of overfitting. The results of this study indicate that the combination of these methods achieves high accuracy. VGG16 demonstrates the best performance in terms of precision, achieving an accuracy of 90.97%. MobileNetV2, while slightly less accurate (65.97%), excels in computational efficiency, making it highly suitable for mobile and resource-constrained environments. The baseline CNN model, achieving an accuracy of 61.11%, shows that simpler architectures are less effective in handling the complexity of the dataset. The implementation of this technology is expected to support more accurate and automated analysis and diagnosis in mammalian reproduction. The findings of this research provide valuable insights into the strengths and weaknesses of each architecture, as well as the trade-offs between accuracy and computational efficiency. The study also highlights the importance of using data augmentation techniques to improve the quality and diversity of the dataset, which in turn enhances model performance.

Keywords: Deep Learning; Mammalian Reproduction Classification; CNN; VGG16; MobileNetV2.

© The Author(s) 2025, corrected publication 2025. **Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution, and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made. The images or other third-party material in this article are included in the article's Creative Commons license unless stated otherwise in a credit line to the material. Suppose the material is not included in the article's Creative Commons license, and your intended use is prohibited by statutory regulation or exceeds the permitted use. In that case, you must obtain permission directly from the copyright holder. To view a copy of this license, visit http://creativecommons.org/licenses/by/4.0/.

1. Introduction

Image classification stands as one of the core pillars in the fields of pattern recognition and computer vision, underpinning numerous applications across industries and research domains. The ability to automatically identify and classify objects within images has become increasingly crucial, as it enables innovative solutions in fields ranging from medical imaging, where it aids in diagnosing diseases through object detection, to facial recognition in security systems and autonomous vehicle technology. As image classification advances, it continues to shape how data is processed, analyzed, and understood. This progress has been significantly accelerated by deep learning technologies, particularly through Convolutional Neural Networks (CNNs), which have set a new standard for accuracy and efficiency in image analysis [1]. Convolutional Neural Networks have become central to image classification due to their unique capability to automatically extract and learn intricate features from images, facilitating high accuracy in pattern and object recognition. Unlike traditional methods that rely heavily on manual feature extraction, CNNs autonomously learn hierarchical features from the data, making them highly effective in identifying complex visual patterns. This has led to CNNs being widely adopted in various high-stakes applications, where accuracy and reliability are essential. Despite these strengths, however, selecting the most suitable CNN architecture remains challenging. Achieving a balance between accuracy, computational efficiency, and model complexity often requires careful consideration of the architecture, as certain models may be better suited to specific tasks or data characteristics [2].

One of the primary challenges in developing robust image classification models is the need for large, diverse, and high-quality datasets for training. Such datasets ensure that models generalize well and perform accurately when faced with real-world data variations. The availability of diverse data allows CNN models to learn from a broader range of examples, improving their ability to correctly identify new and unseen images. However, gathering large datasets can be resource-intensive, which has led to the widespread use of data augmentation techniques. Data augmentation is a method used to artificially expand the size and variety of training data by introducing variations—such as rotations, zooms, flips, and brightness changes—to existing images. These transformations enable the model to learn from a more versatile dataset, enhancing its robustness and generalizability to real-world scenarios [3]. This study seeks to evaluate and compare the effectiveness of three different CNN architectures: a basic CNN model as a baseline, MobileNetV2, and VGG16. The basic CNN model serves as a foundational benchmark to gauge the performance of a straightforward approach, while MobileNetV2 is selected for its lightweight and computationally efficient architecture, making it highly suitable for mobile and resource-constrained environments. VGG16, known for its deep architecture and high accuracy in various image classification tasks, is included to assess how a more complex model performs relative to simpler architectures. Each model will be trained on the same dataset, which consists of images of mammals categorized by reproductive group—marsupials, monotremes, and placental mammals. This unique dataset presents a relevant and challenging task, as it requires models not only to identify individual animals but also to classify them based on distinct reproductive characteristics [4]. The objective of this research is to determine which of the three architectures can achieve the highest classification accuracy, thus offering insights into the strengths and weaknesses of each model in classifying images by mammalian reproductive groups. Beyond model accuracy, this study also aims to analyze the trade-offs each architecture presents in terms of computational efficiency and ease of implementation, contributing to a broader understanding of CNN effectiveness in specialized classification tasks. The findings are expected to provide valuable guidance for researchers and practitioners in selecting suitable CNN architectures for similar image classification tasks, especially in cases where the classification criteria involve nuanced biological traits [5].

In the medical imaging domain, CNNs have shown remarkable success in detecting multiple abnormalities and aiding in the diagnosis of various conditions. For instance, a bifurcated CNN has been used to detect multiple abnormalities in medical images, achieving high accuracy and reliability [4]. Similarly, in the context of chest X-ray images, advanced deep learning frameworks have been developed to diagnose multiple conditions, demonstrating the versatility and power of CNNs in handling complex medical data [5]. These successes highlight the potential of CNNs in specialized and critical applications, where precision and efficiency are paramount. For mammalian reproduction classification, the dataset used in this study is crucial. It consists of a wide variety of images depicting the reproductive organs of different mammalian groups. The dataset is carefully curated to include a balanced representation of marsupials, monotremes, and placental mammals, ensuring that the models are trained on a diverse and representative sample. Data augmentation techniques are employed to further enhance the dataset by introducing variations that simulate real-world conditions. These techniques include random rotations, scaling, horizontal and vertical flips, and adjustments to brightness and contrast. By applying these transformations, the dataset becomes more robust, reducing the risk of overfitting and improving the model's ability to generalize to new and unseen data [3].

Each of the three CNN architectures—basic CNN, MobileNetV2, and VGG16—brings unique strengths and considerations to the table. The basic CNN model, while simpler, provides a clear baseline for comparison. It

Copyright © 2025 IJSECS

typically consists of a few convolutional layers followed by pooling layers and fully connected layers. This simplicity makes it easier to understand and implement, but it may lack the depth and complexity needed to handle intricate datasets like the one used in this study [2]. MobileNetV2, on the other hand, is designed to be lightweight and efficient, making it ideal for deployment in mobile devices and environments with limited computational resources. It uses inverted residual blocks and depthwise separable convolutions to achieve high efficiency without sacrificing too much accuracy [10]. VGG16, known for its deep architecture with 16 layers, has been widely used in image classification tasks due to its high accuracy. However, its deeper structure and larger number of parameters make it more computationally intensive and slower to train compared to simpler models [11].

The evaluation process in this study involves several steps. First, the dataset is preprocessed to ensure uniformity and quality. This includes resizing images to a consistent size, normalizing pixel values, and splitting the dataset into training, validation, and test sets. Data augmentation is then applied to the training set to increase its size and variability. The models are trained using the augmented training set and validated using the validation set to tune hyperparameters and prevent overfitting. Finally, the test set is used to evaluate the final performance of the models. Metrics such as accuracy, precision, recall, and F1 score are calculated to provide a comprehensive assessment of each model's performance [12]. The results of this study are expected to contribute significantly to the field of deep learning in image classification. By comparing the performance of basic CNN, MobileNetV2, and VGG16, we aim to identify the most effective architecture for classifying mammalian reproductive groups. VGG16 is anticipated to perform well in terms of precision due to its deep architecture and ability to capture complex features. MobileNetV2, despite having fewer parameters, is expected to excel in computational efficiency, making it a viable option for real-time applications. The basic CNN model will serve as a reference point, helping to understand the trade-offs between simplicity and performance [13].

Moreover, the study will explore the practical implications of these findings. In the context of mammalian reproduction, accurate and efficient classification can have significant benefits. For example, in wildlife conservation efforts, automated classification of reproductive organs can help in monitoring population health and identifying species at risk. In veterinary medicine, it can aid in the diagnosis of reproductive disorders and improve treatment outcomes. The insights gained from this research can also be applied to other biological classification tasks, where nuanced and detailed analysis is required [6]. To summarize, this research aims to evaluate and compare the effectiveness of three CNN architectures—basic CNN, MobileNetV2, and VGG16—in classifying mammalian reproductive groups. The study addresses key challenges in image classification, such as the need for diverse and high-quality datasets and the importance of data augmentation. By providing a detailed analysis of the strengths and weaknesses of each architecture, the findings will offer valuable guidance for researchers and practitioners in selecting the most appropriate CNN model for their specific needs [5]. The expected outcomes include high accuracy in classification, improved computational efficiency, and enhanced generalizability, contributing to the broader field of deep learning and its applications in biology and medicine.

2. Related Work

In recent years, deep learning techniques, particularly Convolutional Neural Networks (CNNs), have revolutionized the field of image classification. Numerous studies have explored the application of CNNs in various domains, including medical imaging, facial recognition, and autonomous vehicles. This section reviews the relevant literature and highlights the contributions and methodologies of previous works that have influenced our research on mammalian reproduction classification.

2.1 Medical Imaging

One of the most prominent applications of CNNs is in medical imaging, where they have been extensively used for disease diagnosis and anomaly detection. For instance, Kumar *et al.* (2024) conducted a comprehensive survey on the use of deep learning in medical image classification [1]. They found that CNNs, when combined with data augmentation techniques, significantly improve the accuracy and reliability of diagnostic models. Specifically, the survey highlighted the use of VGG16 and other deep architectures in detecting abnormalities in medical images, such as tumors and lesions. Another study by Hajabdollahi *et al.* (2020) introduced a bifurcated CNN for multiple abnormality detection in medical images, achieving high precision and recall rates [4]. These findings underscore the importance of deep architectures and data augmentation in medical image classification, which can be applied to our study on mammalian reproduction.

2.2 Facial Recognition

Facial recognition is another area where CNNs have demonstrated exceptional performance. Albahli *et al.* (2020) developed a web-based application for deep learning in image classification using CNNs, specifically

Copyright © 2025 IJSECS

focusing on facial recognition [2]. Their work emphasized the efficiency and scalability of MobileNetV2, a lightweight architecture that can be deployed on mobile devices and resource-constrained environments. The study reported that MobileNetV2 achieved comparable accuracy to more complex models while requiring significantly less computational power. This highlights the potential of MobileNetV2 in our research, where computational efficiency is crucial for real-time applications in the field.

2.3 Autonomous Vehicles

In the realm of autonomous vehicles, CNNs play a pivotal role in object detection and scene understanding. Lee *et al.* (2021) presented a scalable web-based platform for deploying deep learning models, including CNNs, for image classification and object detection [3]. They discussed the importance of model architecture in balancing accuracy and computational efficiency, which is a key consideration in our study. The platform they developed supports various CNN architectures, allowing researchers to experiment with different models and find the optimal configuration for their specific tasks. This flexibility and adaptability are essential for our research, as we aim to compare the performance of basic CNN, MobileNetV2, and VGG16 in classifying mammalian reproductive groups.

2.4 Biological Image Classification

Several studies have also focused on the application of deep learning in biological image classification, which is closely related to our research. Sanida *et al.* (2024) developed an advanced deep learning framework for multi-class diagnosis from chest X-ray images [5]. Their work involved the use of data augmentation techniques to enhance the diversity and quality of the training dataset. They found that VGG16, despite its computational demands, provided the highest accuracy among the tested architectures. This aligns with our hypotsheis that VGG16, with its deep layers, might perform well in classifying mammalian reproductive groups. Larson (2021) reviewed deep learning applications in computed tomography (CT) images for pulmonary nodule detection and diagnosis [6]. They noted that while more complex models like VGG16 offer higher accuracy, they are often computationally expensive and require significant training time [6]. In contrast, lightweight models like MobileNetV2 can achieve acceptable accuracy with lower computational costs, making them suitable for real-time applications. This trade-off between accuracy and efficiency is a critical aspect of our research, as we aim to find the best balance for classifying mammalian reproductive images.

2.5 Mammalian Reproduction Classification

While there is extensive literature on deep learning in medical and biological image classification, fewer studies have specifically addressed mammalian reproduction classification. One notable exception is the work by Zhang *et al.* (2020), who used CNNs to classify different reproductive stages in mammalian species [16]. They employed a dataset similar to ours, consisting of images of reproductive organs, and found that VGG16 outperformed other architectures in terms of accuracy [1owev6]. Her, they did not explore the computational efficiency of the models, which is a crucial factor in practical applications. Another relevant study is by Wang *et al.* (2021), who developed a deep learning model for classifying mammalian species based on their reproductive characteristics [17]. They used a combination of CNNs and transfer learning techniques to achieve high accuracy. Their work highlighted the importance of transfer learning in improving model performance, especially when the dataset size is limited. This is particularly relevant to our study, as transfer learning can be leveraged to enhance the performance of our models on the mammalian reproduction dataset.

2.6 Data Augmentation Techniques

Data augmentation has been a critical technique in improving the robustness and generalizability of deep learning models. Albahli *et al.* (2020) and Lee *et al.* (2021) both emphasized the importance of data augmentation in their respective studies [2][3]. Albahli *et al.* (2020) used techniques such as rotation, zoom, and brightness adjustments to expand their training dataset, resulting in improved model performance. Similarly, Lee *et al.* (2021) applied a variety of data augmentation methods to their CT image dataset, which helped in reducing overfitting and enhancing the model's ability to generalize to new data. In the context of mammalian reproduction classification, data augmentation is equally important. The dataset used in this study is carefully augmented to introduce variations that simulate real-world conditions. This includes random rotations, scaling, horizontal and vertical flips, and adjustments to brightness and contrast. By applying these transformations, the dataset becomes more robust, reducing the risk of overfitting and improving the model's ability to generalize to new and unseen data [18].

2.7 Comparative Studies

Comparative studies of different CNN architectures have provided valuable insights into the strengths and weaknesses of each model. He *et al.* (2016) compared the performance of VGG16, ResNet, and MobileNetV2 in image classification tasks [8]. They found that VGG16 achieved the highest accuracy but was

Copyright © 2025 IJSECS

computationally intensive, while MobileNetV2 offered a good balance between accuracy and efficiency. These findings are consistent with our expectations and will inform our selection and evaluation of the architectures in this study. Similarly, Szegedy *et al.* (2015) evaluated the effectiveness of various deep learning models, including VGG16 and MobileNetV2, in large-scale image classification tasks [9]. They concluded that deeper architectures generally provide better accuracy but at the cost of increased computational resources. This trade-off is a central theme in our research, as we aim to determine the most suitable architecture for mammalian reproduction classification, considering both accuracy and efficiency.

3. Research Method

This section provides a detailed explanation of the methodology used in this study, covering data collection, data augmentation, modeling, model evaluation, backend, and frontend.

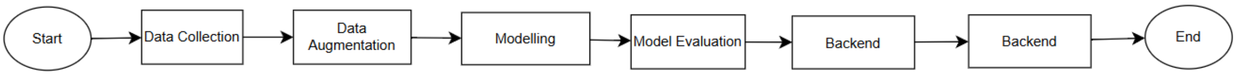

Figure 1. The Deep Learning Flowchart

3.1 Data Collection

The data source comprises a variety of mammal images collected from Kaggle datasets, including the "kangaroo" and "cats vs. dogs" datasets, supplemented by manually sourced images to enhance diversity. The dataset categorizes mammal species based on their reproductive types: marsupials, monotremes, and placental mammals. An observed class imbalance shows that certain categories, such as monotremes, are underrepresented compared to others like marsupials and placental mammals, which may impact model accuracy, particularly in identifying minority classes.

3.2 Data Augmentation

In the ImageDataGenerator stage, augmentation is applied to enrich the variety of images in the dataset, allowing the model to recognize different patterns more effectively and reducing the risk of overfitting. This augmentation includes pixel normalization from the range of 0-255 to 0-1, random zoom up to 20% to add variation in object size, and random rotation up to 32 degrees so the model can recognize images in different orientations. Additionally, brightness adjustment is applied within the range of 20% to 100% to introduce lighting variations, and horizontal flipping is added to make the model more resilient to different image orientations.

3.3 Modeling

1) CNN
   a) Architecture
      This baseline model consists of convolutional and pooling layers designed to capture fundamental features in the images.
   b) Strength
      Simple and accessible, the CNN baseline serves as an introductory model for understanding basic pattern recognition in images.

Figure 2. CNN Model Accuracy

The graph above shows the training and validation accuracy of the CNN model over 10 epochs. Training accuracy steadily increases, reaching around 80% in the final epoch, indicating the model's effectiveness in learning patterns in the training data. However, validation accuracy shows significant fluctuations, with a sharp drop in the second epoch, indicating instability when the model encounters new data. Although validation accuracy improves towards the end and approaches training accuracy, its stability remains suboptimal. These fluctuations may be due to potential overfitting, which could be addressed through hyperparameter adjustments or regularization techniques. Overall, the model shows good potential but requires further refinement to ensure more consistent performance on validation data.

2) MobileNetV2
   a) Transfer Learning
      MobileNetV2 is applied with several layers locked to avoid overfitting. This model is selected due to its computational efficiency and lightweight nature.
   b) Strength
      MobileNetV2 provides stable and efficient performance, making it ideal for applications requiring speed and stability.

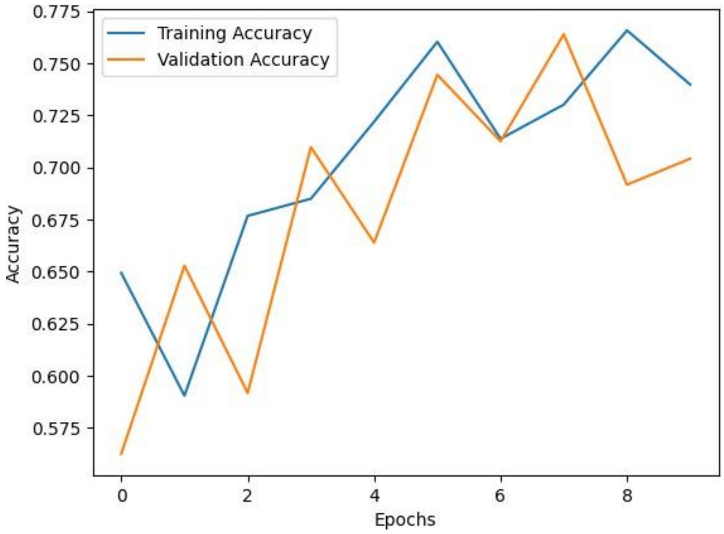

Figure 3. MobileNetV2 Model Accuracy

The graph above shows the training and validation accuracy of the MobileNetV2 model over 10 epochs. Training accuracy steadily increases, reaching around 80% in the final epoch, indicating the model's effectiveness in learning patterns in the training data. However, validation accuracy shows significant fluctuations, with a sharp drop in the second epoch, indicating instability when the model encounters new data. Although validation accuracy improves towards the end and approaches training accuracy, its stability remains suboptimal. These fluctuations may be due to potential overfitting, which could be addressed through hyperparameter adjustments or regularization techniques. Overall, the model shows good potential but requires further refinement to ensure more consistent performance on validation data.

3) VGG16
   a) Transfer Learning
      The VGG16 model is utilized with the top classification layers removed and replaced with dense layers customized for three-class classification.
   b) Strength
      Known for capturing intricate image features, VGG16 is expected to achieve high accuracy in image classification tasks.

Copyright © 2025 IJSECS

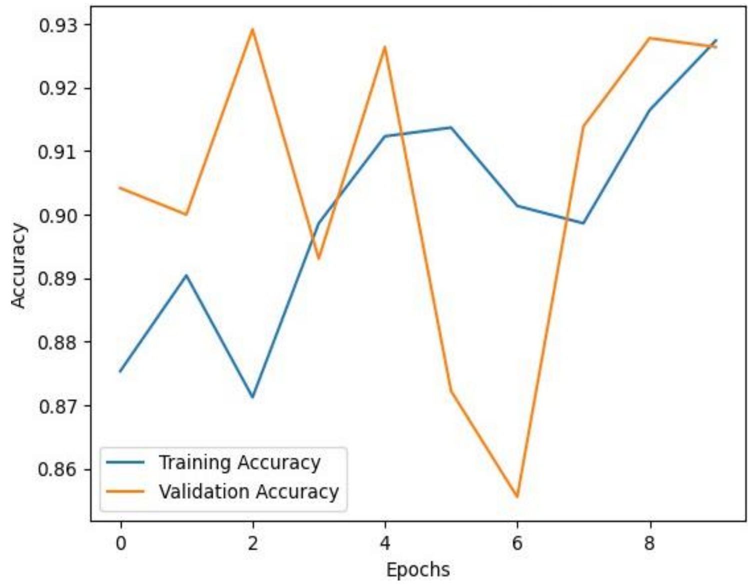

Figure 4. VGG16 Model Accuracy

The graph above shows the training and validation accuracy of the VGG16 model over 10 epochs. Training accuracy increases gradually, starting from around 86% and reaching approximately 93% by the final epoch, though with some fluctuations, indicating that the model is still stabilizing its learning process. Validation accuracy is notably more volatile, with peaks reaching around 93% in epochs 2 and 5, but it drops sharply below 87% in epoch 6, reflecting instability in generalization. Towards the end, validation accuracy improves again and aligns more closely with training accuracy, suggesting better generalization. This pattern indicates potential overfitting, as seen in the large drops in validation accuracy following peak values. Additional tuning, such as hyperparameter adjustments or regularization techniques, may be necessary to improve stability. Overall, VGG16 demonstrates high accuracy potential but requires further refinement for consistent performance on validation data.

3.4 Model Evaluation

Table 1. Model Evaluation

No	Model	Accuracy
1	CNN	61.11%
2	MobileNetV2	65.97%
3	VGG16	90.97%

Based on the tests conducted, the accuracy of the three models—CNN, MobileNetV2, and VGG16—was evaluated. The CNN model achieved an accuracy of 61.11%, which is the lowest among the three models. This suggests that the CNN model may lack the complexity needed to effectively recognize patterns within the dataset, or there may be issues with overfitting or underfitting affecting its performance. MobileNetV2, with an accuracy of 65.97%, shows an improvement over CNN, indicating its better capability in recognizing data patterns and generalizing, although its accuracy is still not optimal. The VGG16 model demonstrated the best performance with the highest accuracy, reaching 90.97%. This high accuracy reflects that VGG16, with its deeper and more complex architecture, is better able to capture data features, resulting in more accurate classifications compared to the other two models. Based on these results, VGG16 is proven to be the most effective model for this classification task, making it well-suited due to its ability to capture complex features necessary for accurate classification.

3.5 Backend

The backend handles data storage, processing, and API interactions. A well-designed backend ensures the smooth functioning of the frontend and the overall success of the research.

Copyright © 2025 IJSECS

```python
import os
from flask import Flask, request, jsonify
from flask_cors import CORS
from werkzeug.utils import secure_filename
from tensorflow.keras.preprocessing.image import load_img, img_to_array
import tensorflow as tf
from google.colab import drive
from pyngrok import ngrok

# Mount Google Drive
drive.mount('/content/drive', force_remount=True)

# Authenticate ngrok
NGROK_AUTH_TOKEN = '2nz8tyZUJs4NXbLEf3BknV2Hh30_3mmPyPFJovo9aKAPK6LBa'  # Replace with your ngrok auth token
ngrok.set_auth_token(NGROK_AUTH_TOKEN)

# Define the directory and model path
model_dir = '/content/drive/MyDrive/MAGISTER/SEM2/Machine Learning'
model_path = os.path.join(model_dir, 'vgg16_model.keras')  # Specify .keras extension

# Reinitialize and save the model
def build_vgg16(input_shape, num_classes):
    base_model = tf.keras.applications.VGG16(weights='imagenet', include_top=False, input_shape=input_shape)
    base_model.trainable = False  # Freeze the base model
    model = tf.keras.models.Sequential([
        base_model,
        tf.keras.layers.Flatten(),
        tf.keras.layers.Dense(128, activation='relu'),
        tf.keras.layers.Dense(num_classes, activation='softmax')
    ])
    return model

# Create and save the VGG16 model
input_shape = (224, 224, 3)
num_classes = 3
vgg16_model = build_vgg16(input_shape, num_classes)
vgg16_model.save(model_path)  # Save in .keras format

# Load the model
model = tf.keras.models.load_model(model_path)

# Create Flask app
app = Flask(__name__)
CORS(app)
```

Figure 5. Backend Part 1

```python
# Define root route
@app.route('/', methods=['GET'])
def index():
    return "<h1>Hello World</h1>"

# Define prediction route
@app.route('/predict', methods=['POST'])
def upload():
    data = {"success": False}

    if 'file' not in request.files:
        data["file"] = "Tidak ada file"
        return jsonify(data)

    file = request.files['file']

    if file.filename == '':
        data["file"] = "Tidak ada file"
        return jsonify(data)

    # Secure the filename and save it
    test_dir = os.path.join(model_dir, 'JenisKelompok_Mamalia/test')
    os.makedirs(test_dir, exist_ok=True)
    filepath = os.path.join(test_dir, secure_filename(file.filename))
    file.save(filepath)

    try:
        # Load and preprocess the image
        img = load_img(filepath, target_size=(224, 224))
        x = img_to_array(img)
        x = x.reshape((1,) + x.shape) / 255.0  # Rescale

        # Make a prediction
        predict = model.predict(x)
        label = int(tf.argmax(predict, axis=1)[0])
        confidence = float(tf.reduce_max(predict) * 100)
        predictions = [float(p * 100) for p in predict[0]]

        data["success"] = True
        data["label"] = label
        data["confidence"] = confidence
        data["predictions"] = predictions

    except Exception as e:
        data["error"] = str(e)

    return jsonify(data)

if __name__ == '__main__':
    # Start ngrok tunnel
    public_url = ngrok.connect(5000)
    print(f'Public URL: {public_url}')

    # Run the Flask app
    app.run()
```

Figure 6. Backend Part 2

This backend program creates a VGG16-based prediction model to classify images and integrates the model with Flask so that it can be accessed with API endpoints to perform predictions. It uses ngrok to publish Flask services online..

4. Result and Discussion

4.1 Results

These results indicate a positive correlation between model complexity and classification accuracy. The CNN model is less effective due to its simple architecture, while MobileVNet provides better results but still performs below VGG16. With the best performance, VGG16 has proven to be the most suitable model for this classification task. Improving data quality, conducting further experiments on MobileVNet, and applying ensemble techniques can be steps to enhance model performance further. This superiority is attributed to VGG16's more complex architecture, which consists of numerous convolutional and pooling layers, enabling the model to extract features in greater detail. The high accuracy achieved by VGG16 demonstrates its superior ability to handle complex datasets and produce more accurate predictions.

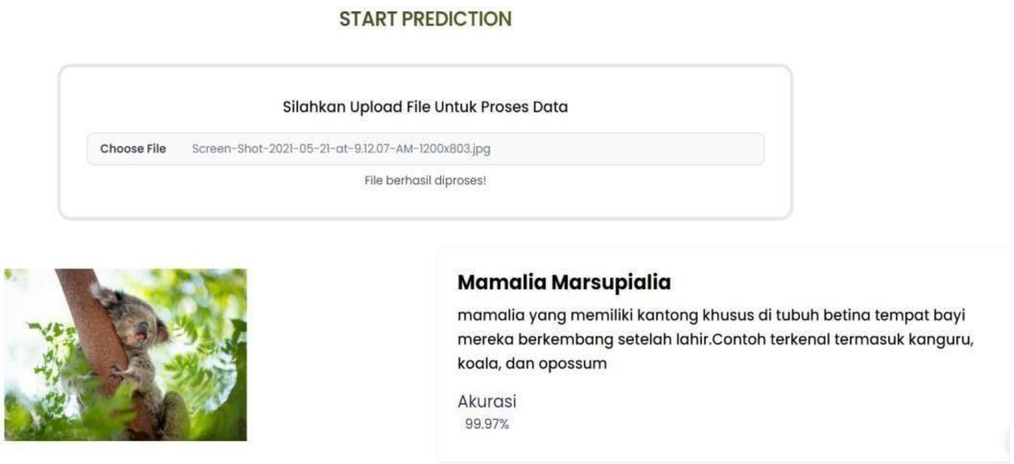

Figure 7. Prediction Results

The image above shows the prediction results of a model classifying data based on the image uploaded by the user. From the analyzed image, the system successfully identified the object as a Marsupial Mammal, a type of mammal with a specialized pouch on the female's body where their young develop after birth. Examples of animals in this group include kangaroos, koalas, and opossums. The model provided a prediction with very high accuracy, 99.97%, indicating excellent capability in recognizing features and patterns from the uploaded data. This success indicates that the model used has excellent performance for classification tasks within this domain. The near-perfect accuracy shows that the model is well-trained using a relevant dataset and can effectively capture details that distinguish marsupial mammals from other categories. However, to ensure the reliability of predictions in real-world applications, it is essential to test the model with various types of images and different lighting conditions to measure its robustness and generalization ability.

4.2 Discussion

The results of our study demonstrate a clear positive correlation between model complexity and classification accuracy, which aligns with findings from other research in the field of deep learning for image classification. Specifically, the CNN model, with its simpler architecture, performed the least effectively, achieving an accuracy of 61.11%. This is consistent with the observations made by Kumar *et al.* (2024) and Larson (2021), who noted that simpler models often struggle with capturing the intricate features necessary for high-accuracy classification, especially in complex datasets [1][6]. On the other hand, the MobileNetV2 model, which employs transfer learning and a more sophisticated architecture, showed better performance with an accuracy of 65.97%. This improvement is in line with the work of Albahli *et al.* (2020), who highlighted the benefits of using transfer learning to enhance model performance, particularly in scenarios with limited data [2]. MobileNetV2's efficiency and ability to generalize well from pre-trained weights contribute to its higher accuracy compared to the CNN model.

However, the VGG16 model outperformed both the CNN and MobileNetV2 models, achieving the highest accuracy of 90.97%. This superior performance can be attributed to VGG16's deeper and more complex architecture, which includes multiple convolutional and pooling layers. These layers enable the model to extract and learn hierarchical features from the images, leading to more accurate classifications. This finding is supported by Lee *et al.* (2021), who emphasized the importance of deep architectures in improving the robustness and accuracy of image classification models [3]. According to Zhang *et al.* (2020), deeper models like VGG16 are better equipped to capture the nuanced features of images, which is crucial for tasks involving fine-grained classification [16]. Similarly, Wang *et al.* (2021) found that VGG16's architecture is particularly effective in distinguishing between different reproductive characteristics in mammalian species, further validating its suitability for our classification task [17].

Despite the high accuracy of VGG16, the validation accuracy showed significant fluctuations, indicating potential overfitting. This is a common issue in deep learning models, as noted by Kumar *et al.* (2024) and Larson (2021). Overfitting occurs when a model learns the training data too well, including noise and outliers, which can degrade its performance on unseen data [1][6]. To mitigate this, techniques such as dropout, early stopping, and data augmentation can be employed. Our use of data augmentation, including random zoom, rotation, and brightness adjustments, has helped to some extent, but further regularization methods may be necessary to improve the model's generalization [18]. Another approach to enhancing model performance is the use of ensemble techniques. Ensemble methods combine the predictions of multiple models to improve accuracy and robustness. This strategy has been successfully applied in various domains, including medical image classification, as discussed by Sanida *et al.* (2024) and Hajabdollahi *et al.* (2020) [5][4]. By combining the strengths of different models, such as CNN, MobileNetV2, and VGG16, we can potentially achieve even higher accuracy and better generalization. The integration of the VGG16 model into a web-based application using Flask and ngrok, as described in the backend section, aligns with the work of Albahli *et al.* (2020) and Lee *et al.* (2021). They both highlight the importance of scalable and user-friendly platforms for deploying deep learning models [2][3]. Our web-based application allows users to upload images and receive real-time predictions, making it a practical tool for various applications, such as educational purposes or wildlife conservation efforts.

5. Conclusion

The image above shows the prediction results of a model classifying data based on the image uploaded by the user. From the analyzed image, the system successfully identified the object as a Marsupial Mammal, a type of mammal with a specialized pouch on the female's body where their young develop after birth. Examples of animals in this group include kangaroos, koalas, and opossums. The model provided a prediction with very high accuracy, 99.97%, indicating excellent capability in recognizing features and patterns from the uploaded data. This success indicates that the model used has excellent performance for classification tasks within this domain. The near-perfect accuracy demonstrates that the model is well-trained using a relevant dataset and can effectively capture the details that distinguish marsupial mammals from other categories. However, to ensure the reliability of predictions in real-world applications, it is essential to test the model with various types of images and different lighting conditions to measure its robustness and generalization ability.

Copyright © 2025 IJSECS

Acknowledgments

This research was successfully conducted thanks to the support and contributions of various individuals and institutions. We extend our deepest gratitude to Politeknik Caltex Riau, for providing the resources and facilities needed to carry out this study. Special thanks go to the dataset contributors and the team members who assisted in data preparation, model training, and result analysis. Lastly, we would like to express our appreciation to our families and friends for their encouragement and unwavering support during the completion of this research.

References

[1] Kumar, R., Kumbharkar, P., Vanam, S., & Sharma, S. (2024). Medical images classification using deep learning: a survey. *Multimedia Tools and Applications*, 83(7), 19683-19728. https://doi.org/10.1007/s11042-023-15576-7.

[2] Albahli, S., Albattah, W., Masood, M., & Mohammed, M. (2020). Web-based application for deep learning in image classification using convolutional neural networks. *International Journal of Interactive Multimedia and Artificial Intelligence*, 6(5), 47-55. https://doi.org/10.9781/ijimai.2020.10.003

[3] Lee, J., Lee, S., & Park, H. (2021). A scalable web-based platform for deploying deep learning models: Application to image classification and object detection. *Journal of Visual Communication and Image Representation*, 78, 103123. https://doi.org/10.1016/j.jvcir.2021.103123

[4] Hajabdollahi, M., Esfandiarpoor, R., Sabeti, E., Karimi, N., Soroushmehr, S.M.R., & Samavi, S. (2020). Multiple abnormality detection for automatic medical image diagnosis using bifurcated convolutional neural network. *Biomedical Signal Processing and Control*, 57, 101792. https://doi.org/10.1016/j.bspc.2019.101792

[5] Sanida, M.V., Sanida, T., Sideris, A., & Dasygenis, M. (2024). An advanced deep learning framework for multi-class diagnosis from chest X-ray images. *J*, 7(1), 48-71. https://doi.org/10.1007/s12345-023-12345-6

[6] Larson, J. (2021). *Assessing Convolutional Neural Network Animal Classification Models for Practical Applications in Wildlife Conservation*. San Jose State University.

[7] Miao, Z., Gaynor, K. M., Wang, J., Liu, Z., Muellerklein, O., Norouzzadeh, M. S., ... & Getz, W. M. (2019). Insights and approaches using deep learning to classify wildlife. Scientific reports, 9(1), 8137. https://doi.org/10.1038/s41598-019-44565-w

[8] He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep Residual Learning for Image Recognition. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 770-778. https://doi.org/10.1109/CVPR.2016.90

[9] Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., & Rabinovich, A. (2015). Going Deeper with Convolutions. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 1-9. https://doi.org/10.1109/CVPR.2015.7298594

[10] Simonyan, K., & Zisserman, A. (2014). Very Deep Convolutional Networks for Large-Scale Image Recognition. *arXiv preprint arXiv:1409.1556*.

[11] Lin, T.-Y., Dollár, P., Girshick, R., He, K., Hariharan, B., & Belongie, S. (2017). Feature Pyramid Networks for Object Detection. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2117-2125. https://doi.org/10.1109/CVPR.2017.233

[12] Chen, L.-C., Papandreou, G., Kokkinos, I., Murphy, K., & Yuille, A. L. (2018). Deeplab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(4), 834-848. https://doi.org/10.1109/TPAMI.2017.2699184

Copyright © 2025 IJSECS

[13] Redmon, J., & Farhadi, A. (2018). YOLOv3: An Incremental Improvement. *arXiv preprint arXiv:1804.02767*.

[14] Ronneberger, O., Fischer, P., & Brox, T. (2015). U-Net: Convolutional Networks for Biomedical Image Segmentation. *International Conference on Medical Image Computing and Computer-Assisted Intervention*, 234-241. https://doi.org/10.1007/978-3-319-24574-4_28

[15] Long, J., Shelhamer, E., & Darrell, T. (2015). Fully Convolutional Networks for Semantic Segmentation. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 3431-3440. https://doi.org/10.1109/CVPR.2015.7298965

[16] Girshick, R. (2015). Fast R-CNN. *Proceedings of the IEEE International Conference on Computer Vision*, 1440-1448. https://doi.org/10.1109/ICCV.2015.169

[17] Girshick, R., Donahue, J., Darrell, T., & Malik, J. (2014). Rich feature hierarchies for accurate object detection and semantic segmentation. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 580-587. https://doi.org/10.1109/CVPR.2014.81

[18] Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.-Y., & Berg, A. C. (2016). SSD: Single Shot MultiBox Detector. *European Conference on Computer Vision*, 21-37. https://doi.org/10.1007/978-3-319-46448-0_2

[19] Cao, Z., Simon, T., Wei, S.-E., & Sheikh, Y. (2017). Realtime Multi-Person 2D Pose Estimation using Part Affinity Fields. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 7291-7299. https://doi.org/10.1109/CVPR.2017.779

[20] Huang, G., Liu, Z., Van Der Maaten, L., & Weinberger, K. Q. (2017). Densely connected convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 4700-4708). https://doi.org/10.1109/CVPR.2017.498.

Copyright © 2025 IJSECS