JIKO (Jurnal Informatika dan Komputer) Accredited KEMENDIKBUD RISTEK, No.105/E/KPT/2022
Vol. 6, No. 3, December 2023, pp. 189-195 p-ISSN: 2614-8897
DOI: 10.33387/jiko.v6i3.7008 e-ISSN: 2656-1948

COMPARISON EFFICACY OF VGG16 AND VGG19 INSECT CLASSIFICATION MODELS

Djarot Hindarto[1], Nihayah Afarini[2], Endah T Esthi H[3]

[1,3]Prodi Informatika, Fakultas Teknologi Komunikasi dan Informatika Universitas Nasional Jakarta
[2]Prodi Sistem Informasi, Universitas Budi Luhur, Jakarta, Indonesia
*Email: [1]djarot.hindarto@civitas.unas.ac.id, [2]nihayahafarini1@gmail.com, [3]endahtriesti@civitas.unas.ac.id

(Received: 23 November 2023, Revised: 27 November 2023, Accepted: 30 November 2023)

Abstract

This research compares two popular deep-learning models, VGG16 and VGG19, for insect classification. This study aims to evaluate insect detection architectures to automate insect identification. We use a large, heterogeneous dataset of insect species, including common pests and beneficial insects, and their images to achieve this goal. The dataset was used to re-adjust the VGG16 and VGG19 models and analyze their classification performance. With an average improvement of 1,8%, VGG19 outperforms VGG16 in insect classification accuracy. VGG19 is more robust because it can handle complex traits and subtle insect morphology differences. Each architecture's model training duration and computational resources are examined for their practicality in real-world scenarios. This research emphasizes deep learning models in insect classification and shows VGG19's higher accuracy and robustness than VGG16. These findings matter to entomologists, agricultural researchers, and pest control experts. They can improve insect identification accuracy and effectiveness using VGG19-based models, which can help solve insect-related problems in various fields. 96.28 percent accuracy was achieved with the implementation of VGG16, according to the experimental findings; 97.07 percent accuracy was obtained with the implementation of VGG19. The research compares VGG16 and VGG19 models for identifying insects, aiming to find the most robust and accurate model for agricultural research.

Keywords: *Insect Classification; Deep Learning; Performance; VGG16; VGG19*

**Corresponding Author: Djarot Hindarto*

1. INTRODUCTION

According to this study, accurate insect classification is crucial to ecosystem preservation and agricultural management. As pests and pollinators, insects affect food webs. Information technology has made insect classification easier, especially with deep learning architectures. Two Convolutional Neural Network architectures [1] are compared for insect identification system improvement. Image processing and deep learning automate classification, saving time and labor. The study improves insect diversity knowledge in agriculture, nature conservation, and entomology. This research advances insect identification systems by using advanced technologies, improving their efficiency and effectiveness in managing diverse insect populations.

In recent times, there has been a notable advancement in insect categorization by utilizing deep learning techniques. Prior studies have employed diverse architectures, such as Convolutional Neural Networks, to conduct insect classification tasks. This study introduces a deep learning model based on convolutional neural networks [2] to detect and classify insects and pests that affect plants. The model employs ensemble methods incorporating VGG16, VGG19, and ResNetv50 architectures. The experimental findings demonstrate that the ensemble model exhibits an accuracy rate of 82.5%, surpassing the performance of prior models and substantiating its efficacy in classifying pests and insects within crops [3]. This study emphasizes the significance of identifying plant diseases in the agricultural sector. The study utilized plant image datasets and implemented deep learning models, including CNN, VGG-16, VGG-19, and ResNet-50. Among these models, ResNet-50 demonstrated the highest accuracy at 98.98%. By utilizing the findings of this study to create web applications for the early detection of plant diseases, this research aids farmers in averting financial losses [4]. Early detection of jute pests is possible with the JutePestDetect model, founded on transfer learning. By employing a diverse set of pre-trained models and utilizing a dataset encompassing seventeen pest classes, the JutePestDetect model demonstrated an impressive accuracy rate of 99%. This strategy can yield substantial advantages for farmers globally [5]. The global COVID-19 pandemic affects many aspects of life. This study proposes a deep learning-based COVID-19 severity model. Advanced pre-processing and feature extraction methods helped the model estimate COVID-19 severity from chest X-rays with 98.2% accuracy [6]. They tested Deep Learning models to detect rice plant pests, diseases, and nutritional deficiencies. The best accuracy was 91.8% for VGG19, while a simple CNN model was 96% for phosphate-deficient leaves. This research can help farmers detect plant issues quickly and non-invasively in precision agriculture [7]. Based on the research mentioned above description, this research has not yielded an improvement in accuracy. Previous studies have reported higher levels of accuracy, with a recorded rate of 99%. However, the present research utilizes the VGG19 model, which achieves an accuracy of 97.07%, aiming to enhance the performance of the VGG16 model, which attained an accuracy of 96.28%. The percentage of research exceeding 96% is higher than that of research that falls below this threshold. The research addresses the discrepancy between 96% and 98% accuracy rates.

This research aims to evaluate the efficacy of VGG16 and VGG19 [8], two insect classification models, in entomology, agriculture, and nature conservation. Identifying insects is essential for comprehending the ecological functions of various species and managing them in natural ecosystems. Convolutional neural networks and deep learning have significantly transformed the field of insect classification through their ability to automate the identification process and efficiently analyze images. The study incorporates various datasets, including detrimental and advantageous insects, to assess the effectiveness of the models in real-life situations. Considering limitations, practical considerations, including training duration and computational resources, will be examined to offer insights for a more extensive implementation. The research project foresees enhanced accuracy and efficiency in categorizing insects, which may have significant ramifications for sustainable agriculture, improved pest control, and advanced entomological studies. The potential benefits of combining entomological knowledge with deep learning technology include improved comprehension and management of insects' various functions in agricultural and ecosystem systems.

2. RESEARCH METHOD

The performance of different deep-learning models in identifying indications of pests, diseases, and nutritional deficiencies in rice plants forms the foundation of this research methodology. We utilize a combined dataset of images from public sources and real-world field captures. This study compares four pre-trained Deep Learning [9] models—InceptionV3, VGG16, VGG19 [10], and ResNet50 [11] — to a more basic Convolutional Neural Network (CNN) model. Experiments were performed utilizing diverse combinations of datasets to assess the accuracy of symptom classification. According to the findings, phosphate deficiency was identified more precisely by the simple CNN model than by VGG19. The outcome of this investigation could be the creation of non-intrusive instruments that aid farmers in rapidly and effectively detecting crop issues.

For this reason, this research proposes a solution to the problem. The following are the research steps as follows:

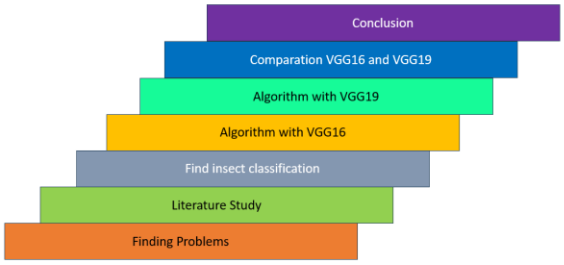

Figure 1. Proposed Research Method
Source: Researcher Property

Figure 1 represents a collection of scholarly investigations that center on addressing issues pertaining to insects in plant species. The initial phase involves conducting a comprehensive review of relevant literature in order to gain a thorough understanding of the detrimental effects of insects on plant growth. The subsequent stage entails advancements in insect classification, with a primary emphasis on the identification of insect species that frequently pose a risk to plant life. In the following phase, the VGG16 and VGG19 deep learning [12] algorithms are employed as a means to classify insects autonomously. The utilization of these two algorithms is intended to address the intricacy and diversity observed in insect morphology, which poses a significant challenge in the process of insect identification on plants. The subsequent focal point pertains to the evaluation of the classification capabilities exhibited by these two algorithms.

The comparison between VGG16 and VGG19 [13] demonstrated that while VGG16 exhibited a satisfactory level of accuracy, VGG19 outperformed it with a superior level of accuracy in the identification and classification of insects. Specifically, VGG19 has demonstrated exceptional capabilities in managing complexity and finer morphological variations in insects, thereby establishing its superiority in the domain of insect identification in plants. Based on the findings of this study, the employment of the VGG19 algorithm in the classification of insects holds significant potential for advancing knowledge and addressing pest-related issues in the realm of plant biology. The efficacy of VGG19 in handling the intricacies of insects instills optimism for the advancement of more efficient strategies in agricultural pest management. This holds the potential for timely detection and implementation of suitable measures to mitigate issues that may adversely impact plant cultivation.

2.1 Convolutional Neural Network

Convolutional Neural Networks [14], [15] (CNNs) are ideal for insect classification because they process image data. CNN applies convolutional layers to insect imagery to simulate human vision and understand intricate features. These layers use filters to extract insect-identifying edges, corners, and textures from the image. After the convolutional layer, CNNs typically have a pooling layer to reduce data dimensionality, a normalization layer to maintain data ranges, and a fully connected layer to classify extracted features. A diverse dataset with labels is needed to train CNNs for insect classification. Iterative weight and bias adjustments reduce prediction errors. This training relies on the backpropagation of errors through layers. CNN [16] ability to handle insect pose, size, and environmental conditions benefits entomology, agriculture, and pest control. Their insect identification accuracy helps these fields and saves time and resources compared to manual methods. CNNs have revolutionized insect classification and greatly benefited fields that rely on insect identification.

2.2 VGG16

The popular VGG16 Convolutional Neural Network [17] model classifies visual objects, especially insects. VGG16's 16 convolutional and image-processing layers made it famous in ImageNet competitions. By extracting edges, lines, textures, and shapes from images, VGG16 classified diverse insect species well. Its adaptability to image dimensions, orientations, and environmental conditions makes it suitable for insect classification in various settings. In entomology and agriculture, VGG16 [18] accurately identifies harmful and beneficial pests. To reduce prediction errors, VGG16 [10] uses insect image datasets to backpropagate neural network weights and biases during training. VGG16 helps scientists and farmers identify, manage, and understand insect ecology in agriculture due to its excellent image recognition performance.

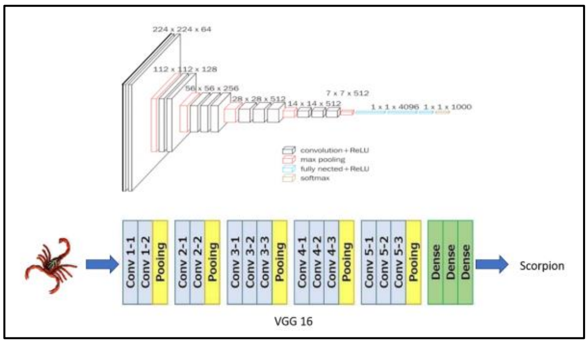

Figure 2. VGG16 Architecture
Source: Google Image Modified

Figure 2 shows the architecture of VGG16, a popular CNN for object classification and image recognition. The model understands complex image features thanks to its 16 convolutions and fully connected layers. Before applying a ReLU activation layer, the image undergoes several 3x3 convolution layers. This layer extracts important image features and patterns. After the convolution layer, the output undergoes max pooling to reduce data dimension while preserving the most prominent elements. This procedure is repeated to protect and enhance image data. Aggregating and linking extracted features to fully connected layers integrate global information from every image. Figure 1 shows an output layer with neurons from each class. Softmax activation functions help generate class probabilities in this layer. The VGG16 architecture uses extracted features to classify images into insect classes. The VGG16 architecture is known for its reliability and complex image recognition tasks. VGG16's high-hierarchical image representation is essential for organizing insects with complex variations despite the model's many parameters. Figure 1 shows the VGG16 architecture's layer structure and connections. This clarifies how the model classifies insect images.

2.3 VGG19

The VGG19 architecture [19], which comprises 19 layers of Convolutional Neural Networks (CNNs), is a noteworthy instrument in insect classification owing to its proficiency in extracting complex features from images of insects (see Figure 3). VGG19, consisting of sixteen convolutional layers and three fully connected layers, distinguishes itself in detecting and identifying various insect species by progressively generating abstract representations via pooling and convolutional layers. The depth of the model improves its capability to accommodate discrepancies in insect sizes, orientations, and contextual environments, thereby enhancing the accuracy of classification. Max pooling and ReLU activation layers [20] are incorporated into the architecture to reduce data dimensions, introduce non-linearity, and improve computational efficiency. The final fully connected layers generate probabilities for various insect classes in an output layer employing a softmax activation function. The computational capabilities and appropriateness of VGG19 for complex image recognition tasks establish it as a prominent option in entomology and agriculture, where it substantially influences the accuracy of insect identification, pest control, and ecological comprehension. The comprehensive illustration of the VGG19 architecture showcases the significance of the initial convolutional layers in capturing fundamental attributes such as shapes and edges. This insight thoroughly comprehends the model's early data progression from input to output. The hierarchical structure of this approach provides the basis for VGG19's ability to differentiate complex insect characteristics, underscoring its pivotal function in insect taxonomy.

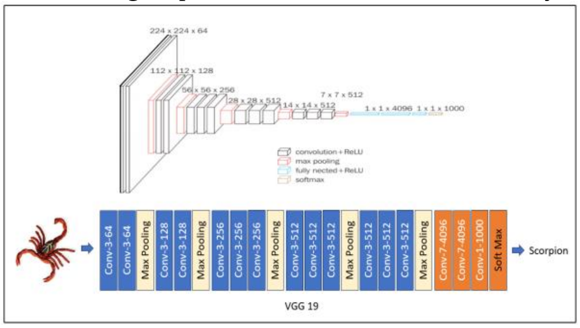

Figure 3. VGG19 Architecture
Source: Google Image Modified

3. RESULT AND DISCUSSION

3.1. Dataset

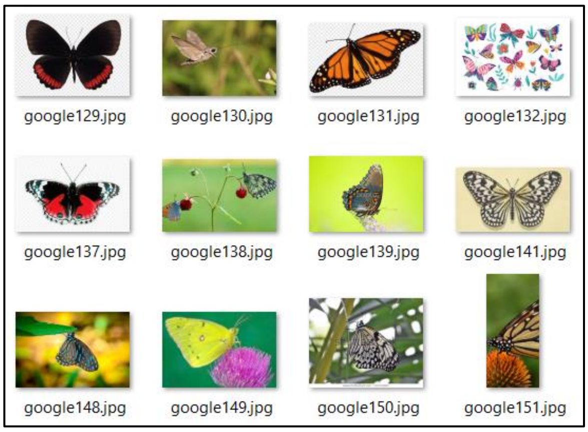

Figure 4. Dataset Insect
Source: Google Image Modified

The dataset presented in Figure 4 comprises insect image data sourced from Google Images. The dataset was acquired through the collection of diverse insect images from publicly available sources on the internet using the Google search engine. The data collection procedure was conducted meticulously to ensure a wide range of insect species, poses, and backgrounds, thereby enhancing the overall representativeness of the dataset. Using insect datasets holds significant value in developing and evaluating insect classification models encompassing intricate VGG19-like architectures. The dataset exhibits a range of diversity that mirrors the ecological conditions insects commonly encounter, including agricultural landscapes, forested areas, and urban environments. Next, the dataset is partitioned into separate training and test subsets to facilitate the model's practical training and evaluate its performance accurately.

By utilizing insect datasets sourced from Google Images, it is anticipated that the resultant model will be able to identify and categorize diverse species of insects accurately. It is essential to consider that these datasets may encompass variations in image quality, lighting conditions, and orientation, posing challenges for models to exhibit robustness and adaptability across diverse circumstances. Datasets play a crucial role in machine learning, particularly in insect classification tasks, as they offer models a comprehensive understanding of the diverse range of insect variations observed in the natural environment. Figure 3 allows researchers and practitioners to follow and comprehend the range of variations that models encounter during the learning process from this dataset. The insect dataset obtained from Google Images, as illustrated in Figure 4, serves a crucial function in advancing and assessing insect classification models while also enhancing our comprehension of insect ecology.

3.2. Experiment VGG16

```
model_VGG16.summary()

Model: "sequential"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
vgg16 (Functional)           (None, 9, 9, 512)         14714688
_________________________________________________________________
global_average_pooling2d (Gl (None, 512)               0
_________________________________________________________________
dropout (Dropout)            (None, 512)               0
_________________________________________________________________
dense (Dense)                (None, 1024)              525312
_________________________________________________________________
dense_1 (Dense)              (None, 5)                 5125
=================================================================
Total params: 15,245,125
Trainable params: 7,609,861
Non-trainable params: 7,635,264
```

Figure 5. VGG16 Architecture

The sequential model under consideration is a neural network architecture rooted in deep learning, comprising layers derived from VGG16. Each layer within this architecture is specifically crafted to facilitate hierarchical feature extraction. The VGG16 architecture incorporates convolutional layers designed to extract intricate patterns from the input data, resulting in an output shape of (None, 9, 9, 512). The model mentioned above has a cumulative parameter count of 14,714,688, enabling it to acquire intricate representations. Following the convolutional layers, the process of average pooling is executed to diminish the spatial dimension, thereby facilitating the consolidation of information. To mitigate the issue of overfitting, a dropout layer is incorporated into the training process, wherein a subset of neurons is deliberately and randomly deactivated. Subsequently, two densely connected layers, called dense and dense_1, are aggregating and abstracting features. The ultimate output layer, comprising 5,125 parameters, generates predictions from the model.

The comprehensive model encompasses 15,245,125 parameters, which can be classified into two distinct categories: trainable and non-trainable. The trainable parameters, which amount to a total of 7,609,861, correspond to the weights and biases that undergo updates throughout the model's training process. In the VGG16 architecture, the non-trainable parameters consist of 7,635,264 fixed parameters. The architectural design, characterized by a substantial quantity of parameters, achieves a harmonious equilibrium between intricacy and efficacy, enabling the model to acquire intricate characteristics from input data while preserving its capacity for generalization. Utilizing a sequential structure facilitates the smooth and effective transmission of information across various hierarchical levels, thereby leading to developing deep learning models that are both efficient and amenable to training.

Figure 6. Performance Accuracy and Loss VGG16

The performance of the VGG16 model is depicted in Figure 6, exhibiting an accuracy rate of 96.28% and a loss value of 0.1141. The model's classification accuracy, which stands at an impressive 96.28%, demonstrates its proficiency in accurately categorizing test data. In the present context, the observed low loss value of 0.1141 indicates the model's predictive accuracy concerning the accurate labels. A low loss value in this framework signifies the model's efficacy in mitigating prediction errors throughout the training phase. The graph depicts the model's stability and accuracy in classification tasks, demonstrating that the VGG16 model can comprehend and generalize intricate data patterns. Consequently, it yields dependable and consistent outcomes in classification endeavors.

Table 1. Training Accuracy and loss VGG16

Epoch	Acc	Val Acc	Loss	Val Loss
25	0.9173	0.9167	0.2421	0.3250
50	0.9415	0.8438	0.1575	0.7384
75	0.9572	0.0885	0.3159	0.9375
100	0.9628	0.9062	0.0929	0.4130

Table 1 shows VGG16 model training results from four epochs. First epoch (25) training accuracy was 91.73%, validation accuracy 91.67%. Although the results are good, the slight difference between training and validation accuracy suggests the model did not overfit early on. The second epoch (50) saw training accuracy rise to 94.15% and validation accuracy fall to 84.38%. This may indicate that the model is adapting too well to training data, which can hurt performance on new data. Training accuracy rose to 95.72% in the third epoch (75), but validation accuracy plummeted to 8.85%. This indicates that the model needs to be more balanced, in which excessive detail is extracted from the training data and fails to represent the new data accurately. In the final epoch (100), training accuracy was retained at 96.28%, and validation accuracy rose to 90.62%. However, the training-validation accuracy gap remains high, indicating difficulties in generalizing the model to new data. During training, loss values match accuracy. Significant loss value changes often mean model behavior with training and validation data. These training results help evaluate and improve model performance, including overfitting and generalization.

3.3. Experiment VGG19

```
model_VGG19.summary()

Model: "sequential_1"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
vgg19 (Functional)           (None, 9, 9, 512)         20024384
_________________________________________________________________
global_average_pooling2d_1 ( (None, 512)               0
_________________________________________________________________
dropout_1 (Dropout)          (None, 512)               0
_________________________________________________________________
dense_2 (Dense)              (None, 1024)              525312
_________________________________________________________________
dense_3 (Dense)              (None, 5)                 5125
=================================================================
Total params: 20,554,821
Trainable params: 9,969,669
Non-trainable params: 10,585,152
```

Figure 7. VGG19 Architecture

Figure 7, the neural network employed in this study is a sequential model that adopts the VGG19 architecture. Each layer of the VGG19 network contributes to the hierarchical extraction of features. The resulting shape of the model's final output is (None, 9, 9, 512), and the total number of parameters in this model is 20,024,384. Following the convolutional layer, the spatial dimension is reduced by implementing average pooling. Subsequently, a dropout layer is employed to mitigate the risk of overfitting. The layer produces an output with dimensions (None, 512), which is later linked to dense layers: dense_2 and dense_3. These dense layers have 525,312 and 5,125 parameters, respectively. The ultimate output layer consists of 5,125 parameters responsible for generating predictions made by the model. The model under consideration encompasses 20,554,821 parameters, which can be further categorized into 9,969,669 trainable parameters and 10,585,152 non-trainable parameters. Trainable parameters contain weights and biases that undergo updates during the training process, whereas non-trainable parameters refer to those parameters that remain fixed within the VGG19 architecture. The architecture possessing notable parameters provides the capability to comprehend and extract intricate features from input data, thereby enabling the development of deep learning models that exhibit robustness and efficacy in classification tasks.

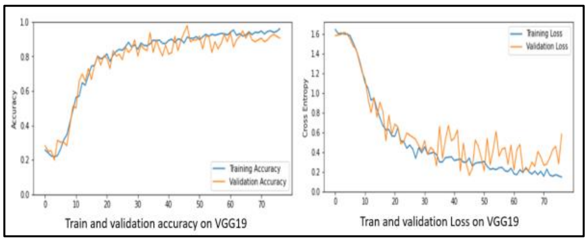

Figure 8. Performance Accuracy and Loss VGG19

The performance of the VGG19 model is illustrated in Figure 8. The model achieved an accuracy of 97.07% and a loss value of 0.1465. The model's notable accuracy rate of 97.07% indicates its capacity to categorize test data accurately. In contrast, the model's accuracy concerning the actual labels is demonstrated by the low loss value of 0.1465, which suggests that the VGG19 model effectively mitigates prediction errors throughout the training phase. The stability and accuracy of the model's classification operations are depicted in this graph, which demonstrates low loss values and high accuracy. These outcomes indicate that VGG19 can comprehend and extrapolate intricate patterns in data. The model's performance indicates that VGG19 is a dependable and precise instrument for classification tasks, as it accurately predicts labels from the test data provided.

Table 2. Training Accuracy and loss VGG19

Epoch	Acc	Val Acc	Loss	Val Loss
25	0.8407	0.8125	0.4389	0.5942
50	0.9143	0.8958	0.2938	0.3423
75	0.9595	0.9271	0.1703	0.4581
100	0.9707	0.9062	0.1465	0.5811

Four main phases of insect classification training are completed in epochs. Initial model accuracy was 84.07% on training data and 81.25% on validation data after 25 epochs. The model's insect identification accuracy is good but can be better. After 50 epochs, the second phase increased accuracy to 91.43% on training data and 89.58% on validation data. This shows that the model is learning insect patterns and features, improving results on new data. After 75 epochs, training data accuracy reached 99.95%, and validation data reached 92.71% in the third phase. This improvement shows that the model can recognize the dataset's insect variety and complexity accurately. In the fourth phase, after 100 epochs, validation accuracy dropped to 90.62%, while training accuracy reached 97.07%. This may indicate overfitting, where the model overfits training data and undergeneralizes validation data. Understanding and overcoming overfitting requires more research. This insect classification training shows how models have evolved to realize insect characteristics, suggesting that entomology and other related fields may use it for insect identification and classification.

3.4. Discussion

The sequential model is a deep learning architecture based on VGG16, a hierarchical feature extraction convolutional neural network. Convolutional layers and average pooling consolidate information in the 14,714,688-parameter model, which outputs (None, 9, 9, 512). A dropout layer randomly deactivates neurons to reduce overfitting. Two densely connected layers, dense and dense_1, abstract features to leave an output layer with 5,125 prediction parameters. The model's 15,245,125 parameters balance complexity and efficacy for robust generalization. Performance shows test data categorization proficiency with 96.28% accuracy and 0.1141 loss. Four epochs of training show overfitting issues.

In contrast, the 20,024,384-parameter VGG19 model uses average pooling and dropout layers. An output layer with 5,125 parameters results from hierarchical feature extraction by dense_2 and dense_3. The model's 97.07% accuracy and 0.1465 loss indicate accurate categorization. With 20,554,821 parameters, the model's performance suggests classification accuracy. Over four phases and 100 epochs, the insect classification training shows the model's evolution in recognizing intricate insect patterns. The drop in validation accuracy suggests overfitting despite high training accuracy. Understanding and mitigating these challenges is essential for deploying the model in entomology and related fields, enabling accurate insect identification and classification.

This study hypothesizes that hierarchical feature learning neural network models like VGG16 and VGG19 can recognize and classify insects from input data. This study aims to develop a model that can effectively discern insects while simultaneously considering the trade-off between complexity and generalizability. This research is vital because rapid and accurate insect identification can improve ecosystem and biodiversity understanding in entomology and related fields. The research results are promising, but many unanswered questions, including model overfitting and generalization to new data. Thus, future research can improve the model to avoid overfitting, explore more advanced neural network architectures, and develop more representative datasets to improve automatic insect identification results. Image processing and artificial intelligence could be used in entomology research to conserve biodiversity and better understand ecosystems.

4. CONCLUSION

The primary objective of this study is to construct and assess neural network models that are founded on the VGG16 and VGG19 architectures. These models will be utilized to identify and categorize insects by leveraging hierarchical features. The findings of the study indicate that these models exhibit a commendable degree of accuracy in the task of insect identification. Nevertheless, it is imperative to address additional concerns such as the issues of overfitting and the ability to generalize to novel data. Subsequent investigations may direct their attention towards enhancing the model's precision, investigating more intricate structures, and constructing datasets that are more representative in order to enhance the dependability of insect identification. The significance of this study resides in its potential to enhance comprehension of ecosystems and biodiversity by means of expedited and precise insect identification. The model exhibits a notable capacity to accurately identify insects, thereby holding significant promise in facilitating entomological research and promoting the conservation of biodiversity. The amalgamation of image processing technology and artificial intelligence has the potential to make a substantial impact on comprehending ecosystems in their entirety and comprehending the significance of insects within these ecosystems. Hence, this study not only facilitates a deeper comprehension of insects but also establishes a basis for the advancement of more efficient insect identification techniques in subsequent research endeavors.

5. REFERENCES

[1] A. Lakhan, T. Grønli, G. Muhammad, and P. Tiwari, “EDCNNS : Federated learning enabled evolutionary deep convolutional neural network for Alzheimer disease detection,” vol. 147, 2023.

[2] C. Methods *et al.*, “ScienceDirect Isogeometric Convolution Hierarchical Deep-learning Neural Network : Isogeometric analysis with versatile adaptivity,” no. xxxx.

[3] S. Masood, “Exploring Deep Ensemble Model for Insect and Pest Detection from Images,” vol. 00, 2023.

[4] M. Islam *et al.*, “DeepCrop : Deep learning-based crop disease prediction with web application,” vol. 14, no. August, 2023.

[5] S. Hasan *et al.*, “Smart Agricultural Technology JutePestDetect : An intelligent approach for jute pest identification using fine-tuned transfer learning,” vol. 5, no. June, 2023.

[6] M. S. Bhupal, Y. M. Rao, C. Venkataiah, G. L. N. Murthy, and M. Dharani, “Deep learning based classification of COVID-19 severity using hierarchical deep maxout model,” vol. 88, no. November 2023, 2024.

[7] B. Dey, M. Masum, U. Haque, R. Khatun, and R. Ahmed, “Comparative performance of four CNN-based deep learning variants in detecting Hispa pest , two fungal diseases , and NPK deficiency symptoms of rice (Oryza sativa),” vol. 202, no. August, 2022.

[8] K. T. N. Duarte *et al.*, “Segmenting white matter hyperintensities in brain magnetic resonance images using convolution neural networks,” vol. 175, no. August, pp. 90–94, 2023.

[9] D. Hindarto, “Performance Comparison ConvDeconvNet Algorithm Vs . UNET for Fish Object Detection,” vol. 8, no. 4, pp. 2827–2835, 2023.

[10] M. Mansour, E. N. Cumak, M. Kutlu, and S. Mahmud, “Deep learning based suture training system,” *Surg. Open Sci.*, vol. 15, no. August, pp. 1–11, 2023, doi: 10.1016/j.sopen.2023.07.023.

[11] D. Hindarto, “Battle Models : Inception ResNet vs . Extreme Inception for Marine Fish Object Detection,” vol. 8, no. 4, pp. 2819–2826, 2023.

[12] E. Suherman, D. Hindarto, A. Makmur, and H. Santoso, “Comparison of Convolutional Neural Network and Artificial Neural Network for Rice Detection,” *Sinkron*, vol. 8, no. 1, pp. 247–255, 2023, doi: 10.33395/sinkron.v8i1.11944.

[13] D. Hindarto, “Comparison of Detection with Transfer Learning Architecture RestNet18, RestNet50, RestNet101 on Corn Leaf Disease,” pp. 41–48.

[14] T. B. Pun, A. Neupane, R. Koech, and K. Walsh, “Detection and counting of root-knot nematodes using YOLO models with mosaic augmentation,” *Biosens. Bioelectron. X*, vol. 15, no. July, 2023, doi: 10.1016/j.biosx.2023.100407.

[15] E. Sze, H. Santoso, and D. Hindarto, “Review Star Hotels Using Convolutional Neural Network,” vol. 7, no. 1, pp. 2469–2477, 2022.

[16] D. Hindarto, “Enhancing Road Safety with Convolutional Neural Network Traffic Sign Classification,” vol. 8, no. 4, pp. 2810–2818, 2023.

[17] D. Hindarto and H. Santoso, “PyTorch Deep Learning for Food Image Classification with Food Dataset,” vol. 8, no. 4, pp. 2651–2661, 2023.

[18] T. Zhang *et al.*, “Weakly-supervised butterfly detection based on saliency map,” *Pattern Recognit.*, vol. 138, p. 109313, 2023, doi: 10.1016/j.patcog.2023.109313.

[19] K. K. Chakraborty, R. Mukherjee, C. Chakroborty, and K. Bora, “Automated recognition of optical image based potato leaf blight diseases using deep learning,” *Physiol. Mol. Plant Pathol.*, vol. 117, no. October 2021, 2022, doi: 10.1016/j.pmpp.2021.101781.

[20] I. Pacal *et al.*, “An efficient real-time colonic polyp detection with YOLO algorithms trained by using negative samples and large datasets,” *Comput. Biol. Med.*, vol. 141, no. September 2021, 2022, doi: 10.1016/j.compbiomed.2021.105031.