HIGH SCHOOL EDITION
Journal of Student Research

Volume 14 Issue 1 (2025)

Pestoscope: AI-Based Pest Detection

Gurman Goraya[1] and Smriti Bhandari[#]

[1]Pine Creek High School, USA
[#]Advisor

ABSTRACT

Before humans became hyper-specialized, most people primarily relied upon farming as a means of food acquisition. However, up to 40% of global crop production is lost to plant pests. With erratic weather linked to global warming, crop-destroying insects have begun thriving in these newfound conditions (Down To Earth 2021). Thus, addressing this challenge has become increasingly urgent to safeguard the livelihoods of millions of farmers and ensure food security for billions. This study aims to develop early pest detection tools that offer significant economic, environmental, and health benefits. We hypothesized that deep neural networks trained through transfer learning could be utilized to detect specific pest types, enabling timely treatment of infestations before they reach critical levels. To build the model, we collected data for six insect pests (bees, moths, slugs, snails, wasps, and weevils) from public datasets available in Kaggle (Marionette 2023) (Pestopia 2023). We applied transfer learning to three convolutional neural networks (CNNs): MobileNetV2 (Sandler et al. 2018), VGG16 (Simonyan and Zisserman 2014), and ResNet50 (He et al. 2016). During training, ResNet50 achieved the highest accuracy of 99.22%, with 20 epochs and a learning rate of 0.001. Subsequently, we added caterpillars to the dataset and retrained the model, resulting in a slightly lower accuracy of 99.00% with the same hyperparameters. The results from these experiments are promising, and with the integration of hardware, we can detect pests early and effectively. This solution offers farmers a proactive way to prevent crop destruction and enhance overall yields.

Introduction

The entire population relies on growing crops as the primary means of sustenance. Nevertheless, agricultural pests hurt much of the world's crop yield. With climate change leading to more inconsistent weather, these destructive insects are flourishing in the now more favorable environment. For example, the total global potential loss due to pests varies from about 50% in wheat, 80% in cotton production, and 31, 37, and 40% for maize, rice, and potatoes, respectively (Oerke 2006). This is costing an estimated value of more than US $470 billion. Further, the losses are considerably higher in the developing tropics of Asia and Africa, where most of the future increase in world population is expected during the next 50 years (Sharma et al. 2017).

Currently, a few species of pests cause the most damage. We targeted these specific pests with the model. Bees, especially invasive species, can aggravate interaction costs and eventually reduce plant reproductive success and crop yield (Aizen et al. 2014). Moths are among the most feared invasive insect pest species. They are the major damaging pests of annual and perennial fiber and food crops, forest products, and stored food commodities throughout the world. The damage inflicted on the fiber and food commodities attacked can be up to 80 percent of yields (Hart 2012). Similar to snails, since the beginning of recorded history, slugs have ravaged crops and today are responsible for billions of dollars in damage. For example, slugs destroyed between $60-$100 million of Oregon's valuable grass seed industry alone (Pokorny 2021). Some estimates find that the total net present cost of wasps is close to $2 billion in economic impacts, with the biggest economic impacts on farming, beekeeping, horticulture, and forestry workers (Macintyre et al. 2015). Most weevils are considered pests and cause environmental damage but some kinds like wheat weevils, maize weevils, and boll weevils are famous for causing huge damage to crops, especially cereal grains

(Mousavi et al. 2022). Most species of caterpillar feed on leaves and young shoots (Mignoni et al. 2022), and their enormous feeding capacity can rapidly decimate a plant (Lu et al. 2023).

To prevent this destruction, current pest elimination methods have been implemented. These methods may include physical measures (cultivation, mechanical weeding, pest exclusion through barriers, traps, etc.) (Vincent et al. 2009), biological pest control (importing predators; planting habitat for natural enemies, etc.), cultural practices (crop rotations to reduce pest incidence; modifying planting/ harvest dates based on pest life cycles; selecting resistant cultivars, etc.), or chemical measures (pesticides such as insecticides, fungicides, and herbicides; pheromones to manipulate pest movement; growth regulators, etc.) (Nazir et al. 2019). Out of all of these pest-control measures, humans often resort to pesticides. Pesticides make a significant contribution to maintaining world food production. In general, each dollar invested in pesticide control returns approximately $4 in crops saved. Estimates suggest that losses to pests would increase by 10% if no pesticides were used at all; specific crop losses would range from zero to nearly 100% (Pimentel 1992). Unfortunately, many of these pesticides are spread over an entire area, including portions of the area not showing symptoms of disease or infestation (Blanket Application 2024).

Regrettably, human pesticide poisonings and illnesses are the highest price paid for pesticide use. There are an estimated 1 million human pesticide poisonings each year in the world, with approximately 20,000 deaths (Pimentel 1992). With the unprecedented growth of the human population, its dependence on crops, and the harmful way that modern society combats pests, it becomes essential to solve this issue more safely.

Currently, some machine learning-based pest detection methods are in use. For example, one DenseNet-enabled IoT-based pest prevention technique uses sound analytics in large agricultural fields to detect pests (Ali et al. 2024). Another uses robotic devices equipped with cameras and sensors that can be used to monitor and find pests (Cubero et al. 2020). However, all of these have serious limitations. First, most of these ideas will struggle with scalability. Considering that farms come in all shapes and sizes, scalability becomes a major consideration. Second, many of the aforementioned methods are highly sensitive to environmental factors. For example, sound propagation and robotic abilities are highly influenced by weather, temperature, and humidity. Third, initial, upfront costs for technology, including sound sensors and robots, can be high, potentially limiting access for small-scale farmers or those in developing regions. Finally, current methods seem to require high levels of technical expertise, and this can be a significant barrier for some users.

To address these challenges, we focused on identifying the most damaging pests and developed a solution using deep learning techniques. By integrating the resulting model with a hardware device, we can detect pests early on, providing a more scalable and user-friendly alternative that is less reliant on environmental conditions and easier to implement for farmers of all sizes.

The model developed brings about a revolution in pest management by providing an accurate, automated system of pest identification. It is, hence, a very promising approach toward mitigating crop damage and enhancing food security with an accuracy rate of nearly 99% and versatile deployment options.

Accuracy and Performance: The highest accuracy reached with the ResNet50-based model was 99.22%, which declined slightly to 99.00% upon the addition of caterpillars within the dataset. The high accuracy rates further ascertain the efficiency of this model for pest detection.

Various Deployment Methods: The model supports two practical deployment methods: a web application with broad accessibility and an IoT device for field use, addressing different user needs and environments.

This study is important for a few key reasons:

1. Economic Impact: The development of early pest detection tools will help farmers evade massive losses of crops, running into billions of dollars annually. This can also be one way to improve global food security and stability. This model can lead to early detection of pests which would reduce broad-scale pesticide applications, reducing the risks of pesticide exposure to human health and the environment. It would allow for the development of safe and environment-friendly methods for pest management, thereby contributing to sustainable agriculture.

2. Deep Learning for Pest Detection: Deep learning for pest detection is one of the most innovative applications of artificial intelligence in agriculture. High accuracy and practical methods of deployment prove that AI is poised to solve pest-related agricultural problems.
3. Accessibility and Practicality: This model has two deployment options: a web application and an IoT device. This makes the model accessible to users all around the world and in very remote agricultural settings. This flexibility enhances the utility and impact of the device across different agricultural contexts.

Methods

We chose to train, validate, and test all models with the Google Collaboratory (Colab) integrated development environment. Colab is a hosted Jupyter Notebook service that requires no setup to use and provides free access to computing resources such as GPUs and TPUs. Colab is especially well suited to machine learning, data science, and education. Additionally, Colab integrates with Google Drive, the cloud-based storage service that we use to store our data, models, and results.

To create all the models, we used multiple machine-learning Python packages that included TensorFlow and Keras. We used TensorFlow to build and train the deep learning models using nodes connected by a flowing network of multidimensional data arrays known as tensors. Keras provided a Google-based deep learning API that supported multiple backend neural network computations. This allowed for quick and effective experimentation. The combination of these tools within Colab made the entire machine-learning process efficient.

For the hardware device, we used an ESP32-WROVER CAM Board to capture real-time images. We chose this board for its ability to process images locally and send data to the cloud for further analysis. This setup allowed for seamless integration of image capture and transmission. The hardware device also included an I2C LCD 1602 module which was connected to the ESP32-WROVER CAM Board for displaying predictions. For the web app, we used a repository powered by GitHub and Streamlit. This additional component enhanced the user experience by providing visual information in a clear and concise manner.

We developed a deep-learning model using three CNN architectures: MobileNetV2, VGG16, and ResNet50. To come up with the most accurate detection model, we conducted transfer learning on three different CNN architectures: MobileNetV2, VGG16, and ResNet50. We conducted transfer learning by removing the top layer from the base model and adding three new layers on top of them. The first layer added is GlobalAveragePooling2D. This layer is similar to other pooling layers used in CNNs to reduce the spatial dimensions of the feature maps by averaging the values in the feature maps. It accepts a 4D tensor and computes an average over all values in each feature map, returning a 2D tensor. It is an operation for simplifying a model by reducing spatial dimensions to a single value per feature map, therefore decreasing the number of parameters and making the network more computationally efficient and less prone to overfitting. The second layer is a fully connected layer with 100 neurons and a 'ReLU' (Rectified Linear Unit) activation function. The final layer is a fully connected neural network layer, with 7 neurons and a 'Softmax' activation function to give out a probability value for different pest classes. Regardless of the probability itself, the model will take the class with the highest probability as the final output.

The process of creating a successful model begins with gathering diverse data from multiple sources. After obtaining data, we split it into train data (2782 images), validation data (330 images), and test data (301 images). Then, we used the train data to train the model by experimenting with 3 types of convolutional neural networks, 5 epoch values, and 7 learning rate values (Figure 1). After we trained, validated, and tested all experiments, we saved a final predictive model. We used the validation dataset during model development to fine-tune hyperparameters and assess how well the model generalizes to unseen data. Validation results only provide feedback for model improvement; they don't guarantee real-world accuracy. The test dataset was a separate subset of data that the model had never seen before. It serves as an unbiased estimate of the final model's performance. By evaluating the model on the test data, we confirmed its effectiveness in real-world scenarios.

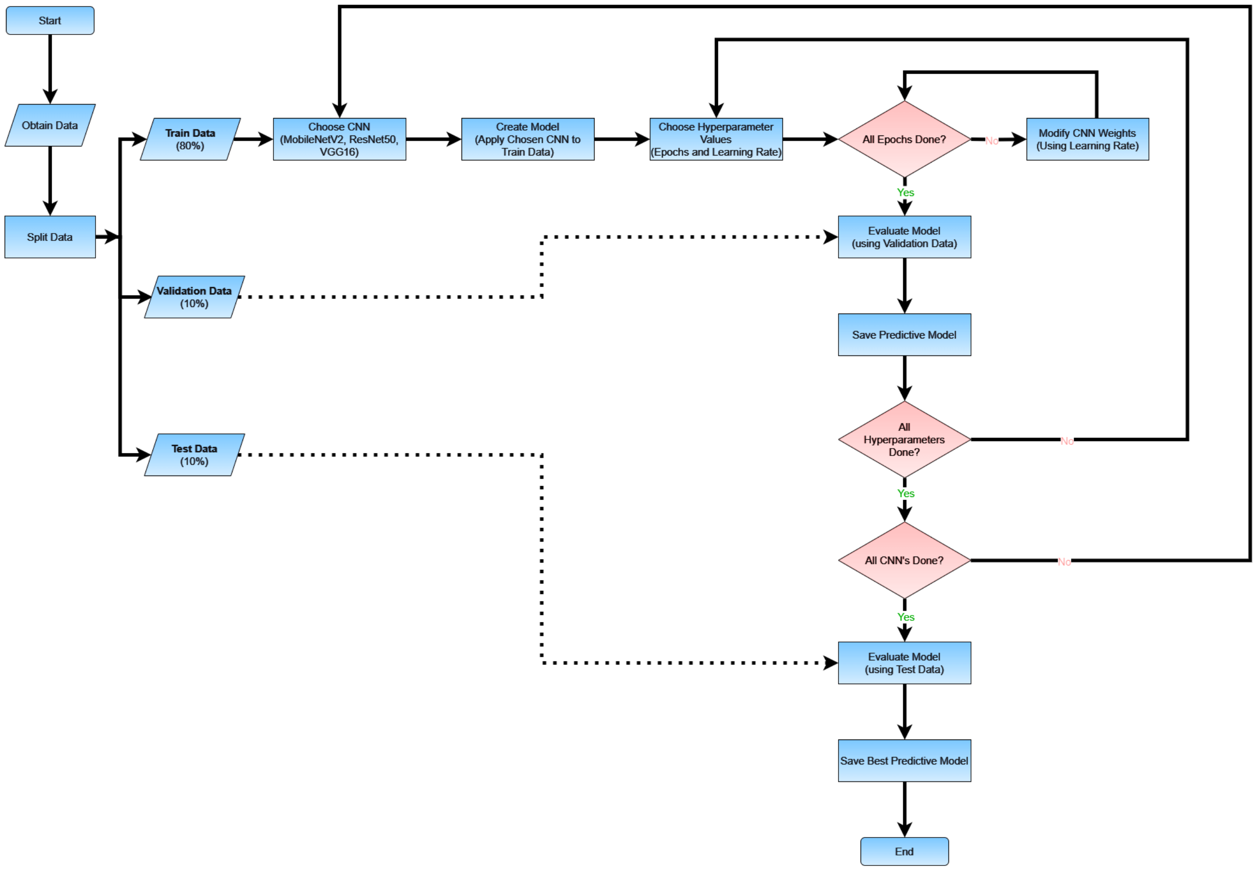

Figure 1. Flow chart for the process of determining the best model. Flow chart illustrating the process used to determine the best model for each CNN based on the accuracy. We obtained data and then divided it into three categories: train, validation, and test. We then trained the model using the train data by experimenting with three different convolutional neural network types, five different epoch values, and seven different learning rate values. Throughout the model's development process, we utilized the validation data to evaluate the model's generalizability to new data and adjust hyperparameters. Then, we evaluated the model's accuracy using the unseen test data. Once all experiments have been trained, verified, and tested, we save the best final prediction model.

Results

The best way to combat the pest and pesticide problem is to detect the type of pest to target and eventually treat the infestation before the infestation reaches unstoppable levels. Deep learning can automate this process and guarantee safety and efficiency.

We experimented with three CNN architectures: MobileNetV2, VGG16, and ResNet50, utilizing pre-trained ImageNet weights for transfer learning. The experiments were conducted by varying hyperparameters such as learning rates and epochs, with performance measured using accuracy values for comparison. The Epochs varied between 10 and 50 while the learning rate varied between 0.000001 and 0.05.

To observe the performance of the different models while training, we generated multiline plot graphs for all of the experiments. The learning rate was on the x-axis and ranged from very small (0.000001) to larger values (0.1). A smaller learning rate means slower convergence during training and a larger epoch value, while a larger learning rate may lead to overshooting and instability. The y-axis represents accuracy which measures how well the machine learning model performs on the given data. Higher accuracy values indicate better model performance. Each colored line corresponds to a specific number of training epochs (10, 20, 30, 40, and 50). As the number of epochs increases, the model learns more from the data, potentially improving accuracy. Generally, accuracy increases with a larger epoch value, but there are diminishing returns. Optimal performance lies within a range of epochs. This graph provides valuable insights for tuning machine learning models, emphasizing the balance between learning rate and epochs to achieve optimal accuracy.

In addition to the graph, we also used confusion matrices to determine the best model. A confusion matrix helps evaluate the performance of a classification model, showing where it excels and where it makes errors. The diagonal cells represent correct predictions, while off-diagonal cells indicate misclassifications. Darker shades correspond to higher values. Confusion matrices can have true positives (the number of instances correctly predicted as a specific class), false positives (the number of instances incorrectly predicted as a specific class), true negatives (the number of instances correctly predicted as not belonging to a specific class), and false negatives (the number of instances incorrectly predicted as not belonging to a specific class).

All the models performed similarly under the same hyperparameter condition. All epoch values performed similarly under the same learning rate. The optimal learning rate for each model was moderate regardless of epochs. This balances the speed of learning and the stability of convergence. Lower learning rates such as 0.000001 lead to low accuracy, likely due to the slow weight updating. This leads to underfitting where the model fails to capture the underlying patterns in the data. On the other hand, larger learning rates such as 0.05 also lead to lower accuracy. This is due to the model making large updates to the weights, causing it to overshoot the optimal values and potentially leading to instability and divergence.

MobileNetV2's optimal learning rate was 0.005 (Figure 2). The epoch values were consistent while the learning rate led to a rise, plateau, and then decline in accuracy. MobileNetV2's peak accuracy was 78.12%. Although MobileNetV2 struggled with slugs the most, the model's confusion matrix has a clear diagonal feature which makes it evident that MobileNetV2 did not struggle with one specific pest but rather had multiple incorrect predictions for each pest (Figure 3).

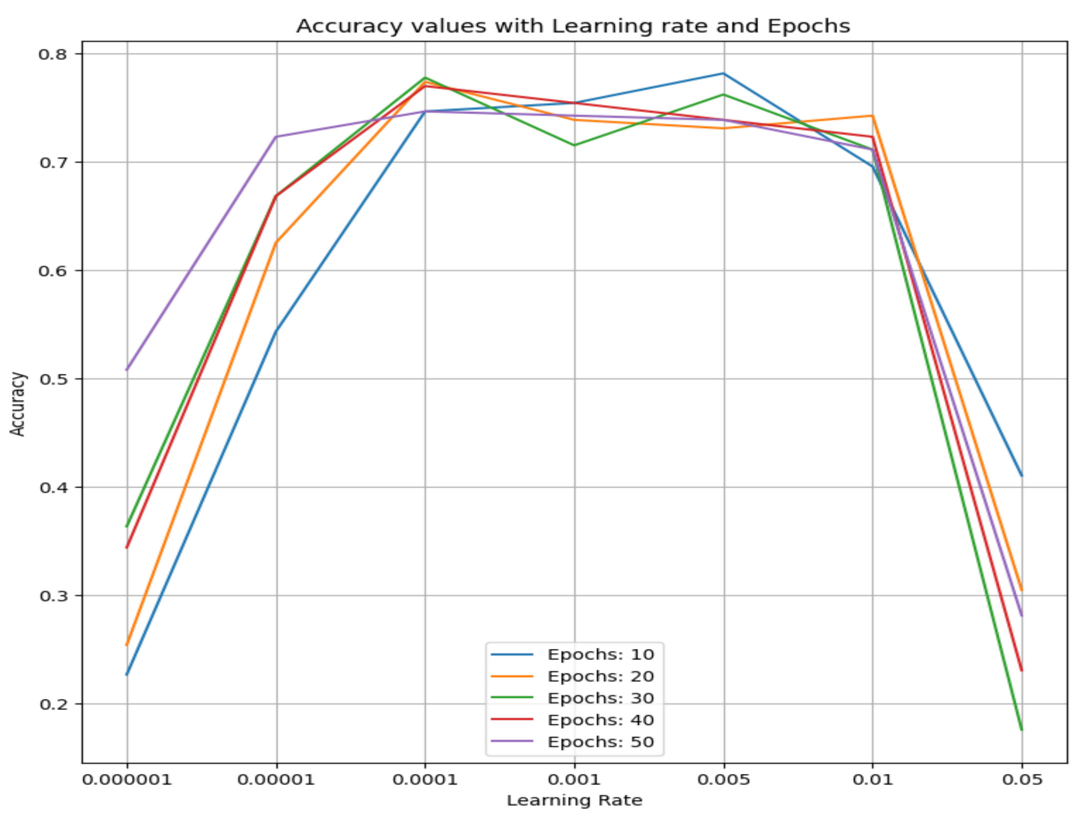

Figure 2. Learning rate vs. accuracy across different epochs for MobileNetV2. Line graph illustrating the relationship between learning rates and accuracy over various epochs. The best model is achieved with a learning rate of 0.005 and 10 epochs. Each epoch is represented by a distinct colored line, demonstrating the impact of learning rate adjustments on the accuracy of a machine learning model during training.

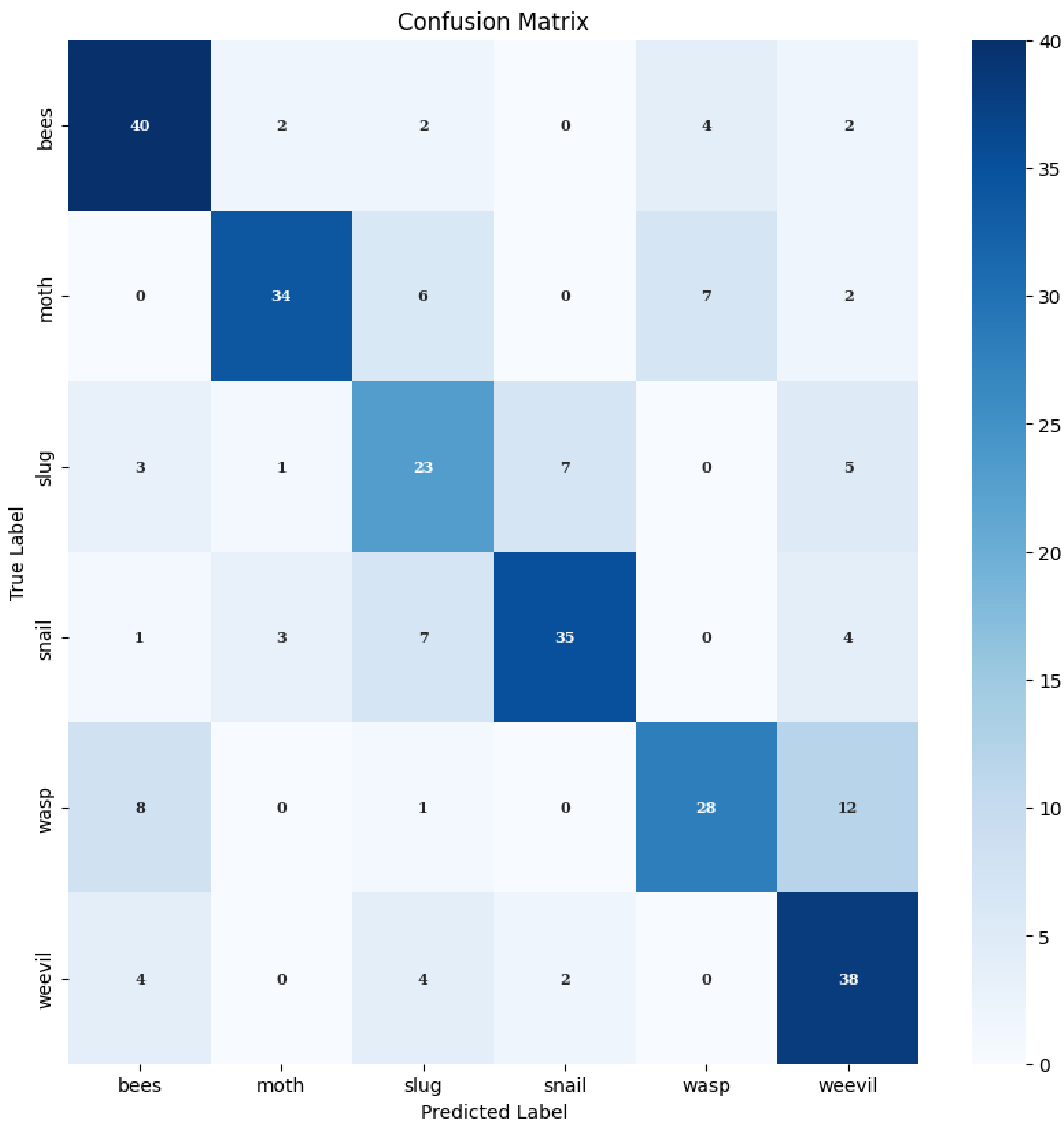

Figure 3. Test confusion matrix for MobileNetV2's best model (learning rate = 0.005, epochs = 10). A detailed confusion matrix displaying the true label versus predicted label accuracy for the categories of bee, moth, slug, snail, wasp, and weevil. The matrix highlights the precision of the model's predictions, with values indicating the number of correct and incorrect predictions for each category.

VGG16's optimal learning rate was 0.01 (Figure 4). Just like the other models, the epoch values were consistent while the learning rate led to a rise, plateau, and then decline in accuracy. VGG16's peak accuracy was 98.83%. VGG16's confusion matrix has a distinct diagonal feature (Figure 5). VGG16 did much better than MobileNetV2 because it did not make multiple incorrect predictions for each pest. Although VGG16 struggled to identify slugs, the model correctly identified all bees in the validation dataset (Figure 5).

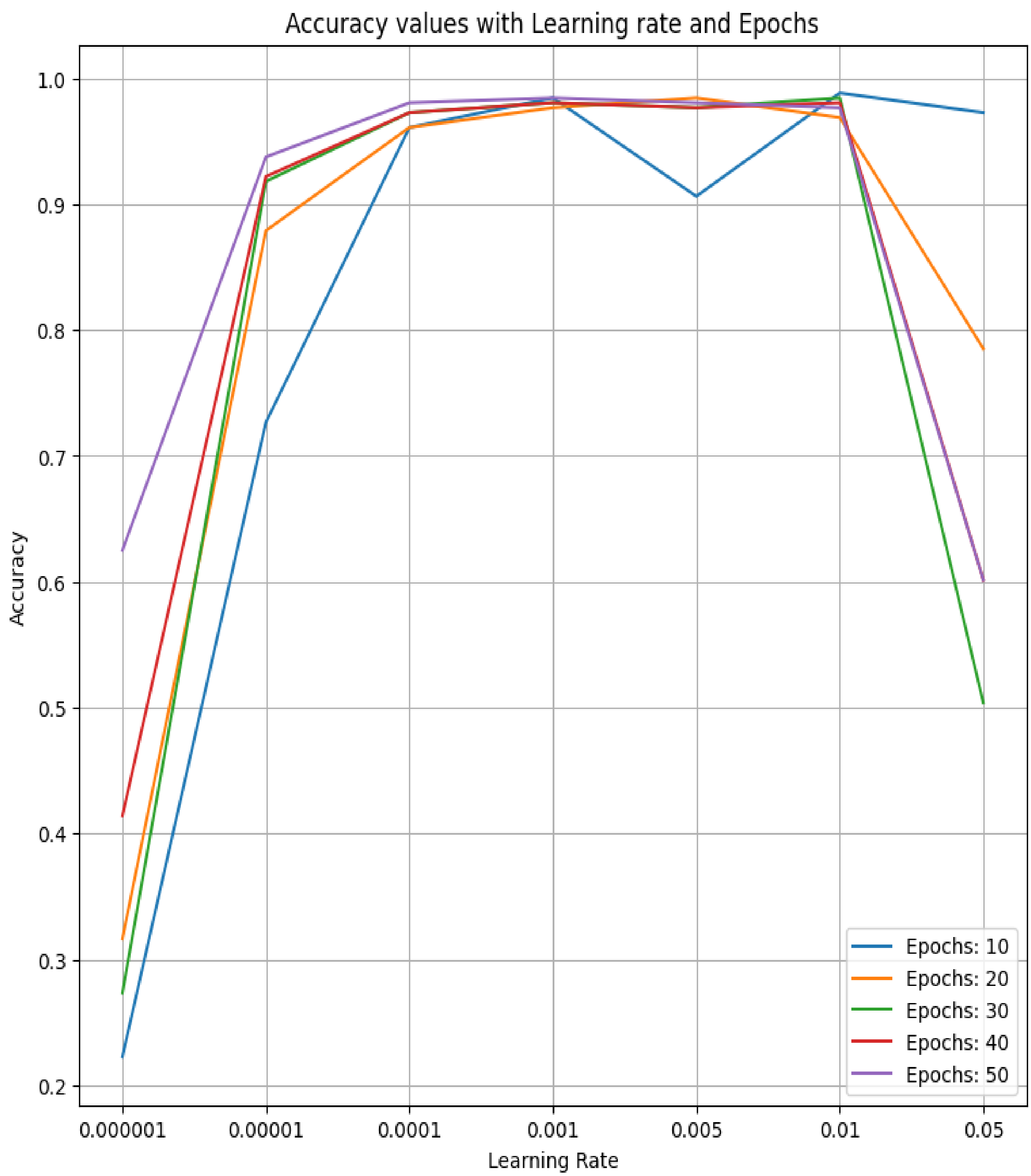

Figure 4. Learning rate vs. accuracy across different epochs for VGG16. Line graph illustrating the relationship between learning rate (0.000001, 0.00001, 0.0001, 0.001, 0.005, 0.01, 0.05) and accuracy (0.2 to 1.0) over various epochs (10, 20, 30, 40, 50). Based on the accuracy, the best model (learning rate = 0.01, epochs = 10) can be determined. Each epoch is represented by a distinct colored line, demonstrating the impact of learning rate adjustments on the accuracy of a machine learning model during training.

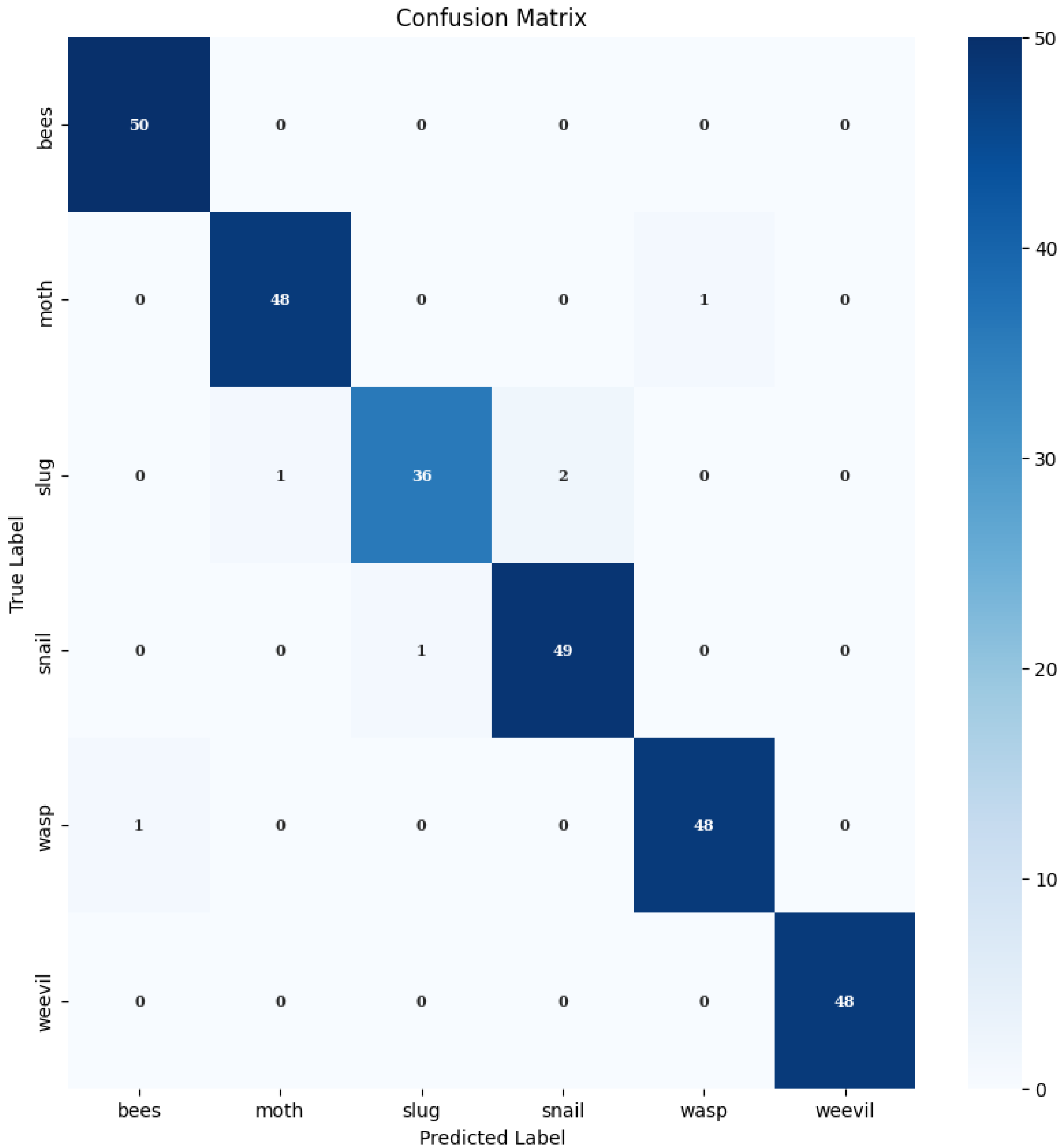

Figure 5. Test confusion matrix for VGG16's best model (learning rate = 0.01, epochs = 10). A detailed confusion matrix displaying the true label versus predicted label accuracy for the categories of bee, moth, slug, snail, wasp, and

weevil. The matrix highlights the precision of the model's predictions, with values indicating the number of correct and incorrect predictions for each category.

ResNet50's optimal learning rate was 0.001 (Figure 6). Similar to the other models, the epoch values were consistent while the learning rate led to a rise, plateau, and then decline in accuracy. ResNet50's peak accuracy was 99.22% with six classes and 99.00% with seven classes. ResNet50's confusion matrix also has a clear diagonal feature (Figure 7a). ResNet50 did much better than MobileNetV2 because it did not make multiple incorrect predictions for each pest. It did similar to VGG16. Although ResNet50 also slightly struggled to identify slugs, the model correctly identified all bees and all snails in the validation dataset (Figure 7a).

The addition of the caterpillar also yielded similar results. The model continued to struggle with slugs but correctly predicted all caterpillars (with slightly fewer validation images) making the category a successful addition (Figure 7b).

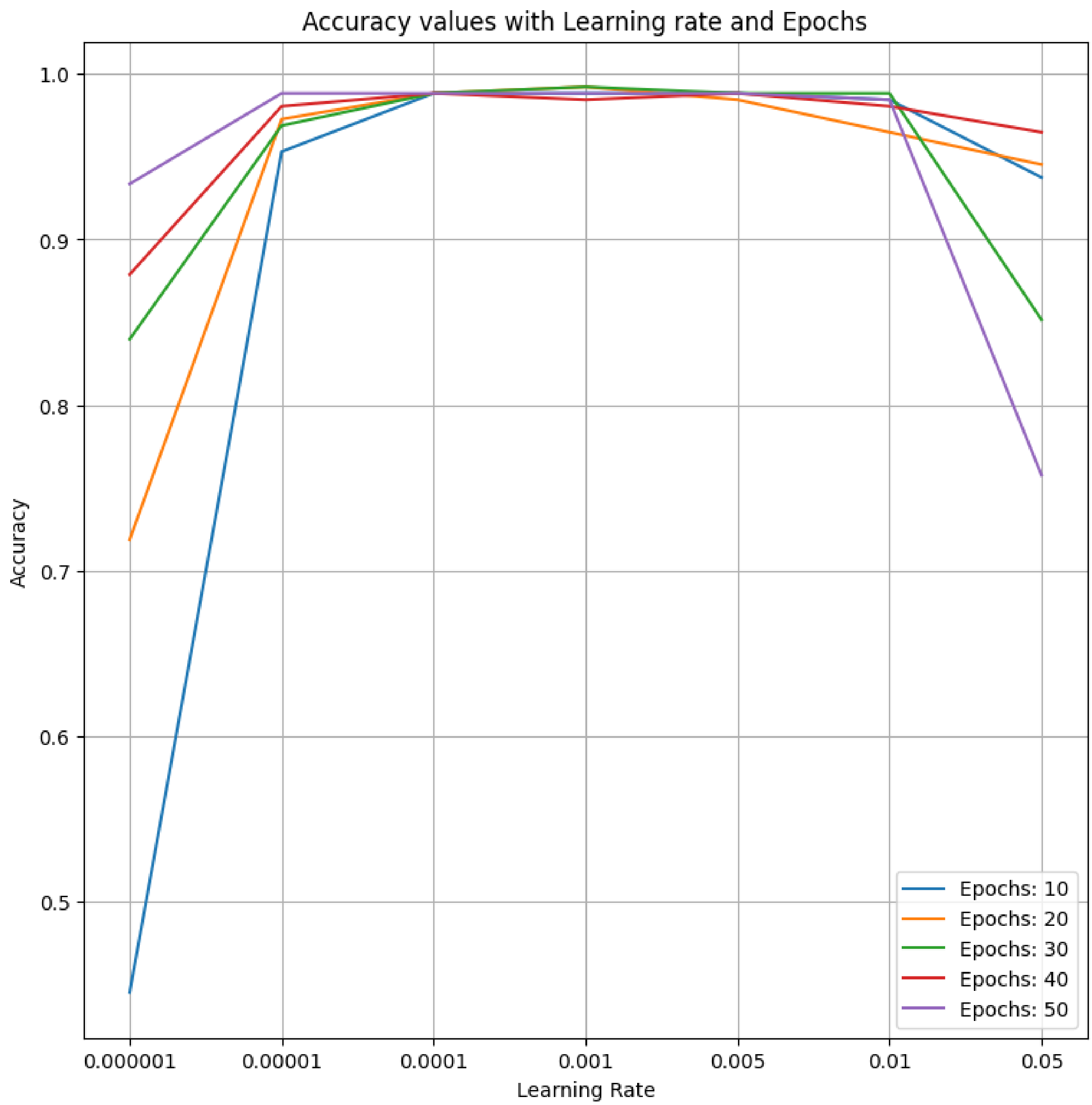

Figure 6. Learning rate vs. accuracy across different epochs for ResNet50. Line graph illustrating the relationship between learning rate and accuracy over various epochs. Based on the accuracy, the best model (learning rate = 0.001, epochs = 20) can be determined. Each epoch is represented by a distinct colored line, demonstrating the impact of learning rate adjustments on the accuracy of a machine learning model during training.

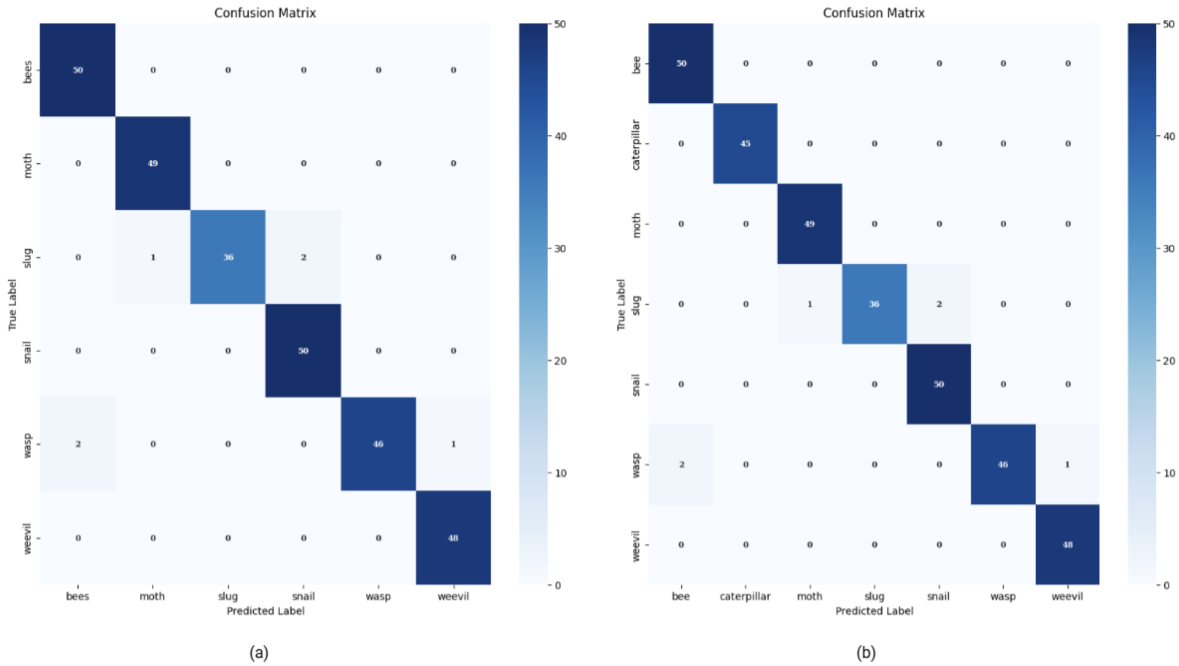

Figure 7. Confusion matrices for ResNet50's best model (learning rate = 0.001, epochs = 20).

(a) Number of classes = 6 (bees, moths, slugs, snails, wasps, and weevils)

(b) Number of classes = 7 (bees, caterpillars, moths, slugs, snails, wasps, and weevils)

We identified that the optimal configuration was ResNet50 with an epoch of 20 and a learning rate of 0.001, achieving a peak accuracy of 99.22%. We further extended this model to include the identification of caterpillars, resulting in an accuracy of 99.00%.

After experimenting with all three CNN architectures, the best ResNet50 model was deployed in both a web application and an IoT device consisting of an ESP32-WROVER CAM Board along with an LCD 1602 Module.

Discussion

The project aimed to develop a machine capable of identifying pests from images to facilitate appropriate pest control measures. The research employed deep learning CNNs like MobileNetV2, VGG16, and ResNet50, varying hyperparameters such as learning rate and epochs to optimize model performance. We identified that the optimal configuration was ResNet50 with epochs set to 20 and a learning rate of 0.001, achieving a peak accuracy of 99.22%. We enhanced this model to include the identification of caterpillars, resulting in an accuracy of 99.00%. This confirms the hypothesis that deep learning techniques can detect the pest type to treat the infestation before it reaches unstoppable levels.

After the ResNet50 model reached an accuracy of 99.22% with 20 epochs and a learning rate of 0.001, we identified the need for another category. We added the caterpillar as an extra category for pests that the trained neural network had to be able to classify. This was possible because of the very high performance of the ResNet50 model and its versatility to handle the new pest category addition without a large loss in accuracy.

We added caterpillar images, re-trained the model, and tested to produce a slightly lower accuracy of 99.00%. Despite the decrease in accuracy, we consider the model sufficiently highly applicable for practical use. By adding the caterpillar as a known pest, it has further generalized the applicability of the model in assisting farmers and other agricultural actors in detection and management strategies against pests.

With the model ready, we developed two distinct ways to access the model that reflect the needs and abilities of users all around the world.

The first technique is the web application. The web application gives users the ability to interact with the model easily without the need for specialized software or hardware; it just requires internet access. The web app provides a friendly user interface where one can feed in images and view results within seconds. Its ease of use and efficiency make the model very accessible to a large group of people with different levels of technical expertise. Anyone around the world can access the web application at https://pestoscope.streamlit.app.

The second technique is the IoT device. In this integrated device, an ESP32-WROVER CAM Board is used to access the model with the prediction label displayed on an LCD 1602 Module. One can easily integrate this device into existing systems for seamless implementation, making it a portable, convenient, and viable option for a variety of agricultural applications, especially in the field itself.

We used various hyperparameters, such as epochs and learning rates, to train each deep-learning model. ResNet50 turned out to be the best model, with a tuned setup of 20 epochs and a learning rate of 0.001, which reached the highest accuracy of 99.00%. This finding shows the model's ability to very easily distinguish between the classes of pests represented in the dataset, which included bees, caterpillars, moths, slugs, snails, wasps, and weevils.

Another critical aspect was the choice of hyperparameters, particularly the number of epochs and the learning rate. With 105 different combinations of base network, learning rate, and epochs, we followed an iterative process to test and finetune the hyperparameters. This created a balance in model performance with training stability. Further experimentation with different epoch values and learning rates may refine the model's performance even further.

The possibility of detecting pests at such a high degree of accuracy allows for early intervention which is important in minimizing crop loss and preventing further spread of any type of infestation. Since farmers will now be able to identify pests precisely, it will be possible to come up with specific control measures against them, thus cutting down on broad-spectrum pesticides and associated environmental and health risks.

These two model deployment options (web application and IoT device) enhance the accessibility and practical utility of the model. The web application provides a user-friendly interface, hence making the model accessible to many users, while the IoT device gives a more portable solution that is easy to use in the field. This therefore means that the deployment methods ensure that the model can be effectively used within different agriculture settings, whether they are remote fields or urban farms.

Other future research and development that we may pursue include: increasing the dataset by a larger number of pests and pest life stages to increase model accuracy and applicability and using further advanced image augmentation and state-of-the-art architectures to improve the model's performance.

We can also integrate this model with real-time monitoring systems and field sensors to gain continuous surveillance of pests, thus sending automated alerts. Also, studies on the usage of the model within different agricultural contexts and regions will help alter this system to particular pest challenges and environments.

Finally, this deep learning-based pest detection model’s development marks a great improvement in the realm of pest management. With a peak accuracy of 99.00%, the model has versatile deployment options for carrying out the globally challenging task of agricultural pest control. This work shows how artificial intelligence can benefit agriculture, food security, and sustainable pest management strategies. Further work will focus on improving the models and generalizing them for applications that help farmers and other actors in the value chain worldwide.

Conclusion

This project aims to create an image-based automatic pest detection machine for effective pest control. The pests that can be detected include bees, caterpillars, moths, slugs, snails, wasps, and weevils. Some of the deep-learning CNN models compared in this project include VGG16, MobileNetV2, and ResNet50. The performance of the model was enhanced by tuning several hyperparameters, which included learning rate and epochs. After extensive experimentation, we realized that ResNet50 with 20 epochs and a learning rate of 0.001 gave us the best accuracy of 99.00%.

Accuracies of detection at such a high level allow for early intervention, which is necessary for reducing crop loss and the spread of the infestation. With precise identification of pests, farmers will be able to adopt effective measures for targeted pest control, thus reducing dependence on pesticides with blanket measures and minimizing environmental and health risks.

The deep learning-based pest detection model proposed in the present study represents a massive step forward in the field of pest management. This model is considered a flexible option for its deployment to address the global challenge arising in agricultural pest control.

Limitations

Although high accuracy is achieved, several limitations could still influence the performance of the model. No matter how comprehensive this dataset is, it may not cover all kinds of variability that a pest may express, which includes different life stages or variants of a particular pest. More diversified images of pests added to the dataset will increase its generalizability.

There was a slight reduction in accuracy by adding caterpillars to the dataset, proving that even if it adapts to new categories, adding new classes introduces new complexity that affects model performance.

Acknowledgments

I would like to thank my advisor for the valuable insight provided to me on this topic.

References

Down To Earth. At Least 40% Global Crops Lost to Pests Every Year: FAO. Down To Earth. www.downtoearth.org.in/news/agriculture/at-least-40-global-crops-lost-to-pests-every-year-fao-77252.

Marionette. (2023) Agricultural Pests Image Dataset. Kaggle. www.kaggle.com/datasets/vencerlanz09/agricultural-pests-image-dataset.

Pestopia: Indian Pests and Pesticides Dataset. Kaggle, 19 Mar. 2023, www.kaggle.com/datasets/shruthisindhura/pestopia.

Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., & Chen, L. C. (2018). Mobilenetv2: Inverted residuals and linear bottlenecks. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 4510-4520). https://doi.org/10.48550/arXiv.1801.04381.

Simonyan, K., & Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*. https://doi.org/10.48550/arXiv.1409.1556

He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 770-778). https://openaccess.thecvf.com/content_cvpr_2016/html/He_Deep_Residual_Learning_CVPR_2016_paper.html

Oerke, E. C. (2006). Crop losses to pests. The Journal of Agricultural Science, 144(1), 31-43. https://doi.org/10.1017/S0021859605005708.

Sharma, S., Kooner, R., & Arora, R. (2017). Insect pests and crop losses. Breeding insect resistant crops for sustainable agriculture, 45-66. https://doi.org/10.1007/978-981-10-6056-4_2.

Aizen, M. A., Morales, C. L., Vázquez, D. P., Garibaldi, L. A., Sáez, A., & Harder, L. D. (2014). When mutualism goes bad: density-dependent impacts of introduced bees on plant reproduction. https://doi.org/10.1111/nph.12924.

Hart, N. (2012). Suppressing Moth Pests. IAEA, www.iaea.org/newscenter/news/suppressing-moth-pests.

Pokorny, K. (2021). Slugs and Snails, Destructors of Crops and Gardens, Could Be Controlled by Bread Dough. Phys.Org, phys.org/news/2021-08-slugs-snails-destructors-crops-gardens.html.

Macintyre, P., & Hellstrom, J. (2015). An evaluation of the costs of pest wasps in New Zealand. New Zealand. Department of Conservation and Ministry for Primary Industries, Wellington, NZ, 50. https://www.doc.govt.nz/globalassets/documents/conservation/threats-and-impacts/animal-pests/evaluation-pest-wasps-nz.pdf.

Mousavi, S., & Mirinezhad, S. (2022). Weevil damage optimization algorithm and its applications. Journal of Future Sustainability, 2(4), 133-144. https://doi.org/10.5267/j.jfs.2022.10.003.

Mignoni, M. E., Honorato, A., Kunst, R., Righi, R., & Massuquetti, A. (2022). Soybean images dataset for caterpillar and Diabrotica speciosa pest detection and classification. *Data in Brief*, *40*, 107756. https://doi.org/10.1016/j.dib.2021.107756.

Lu, H., Lyu, B., Tang, J., Wu, Q., Wyckhuys, K. A., Le, K. H., ... & Zhang, Q. (2023). Ecology, invasion history and biodiversity-driven management of the coconut black-headed caterpillar Opisina arenosella in Asia. *Frontiers in Plant Science*, *14*, 1116221. https://doi.org/10.3389/fpls.2023.1116221

Vincent, C., Weintraub, P., & Hallman, G. (2009). Physical control of insect pests. In *Encyclopedia of insects* (pp. 794-798). Academic Press. https://doi.org/10.1016/B978-0-12-374144-8.00209-5

Nazir, T., Khan, S., & Qiu, D. (2019). Biological control of insect pest. *Pests Control and Acarology*, *21*. https://books.google.co.in/books?hl=en&lr=&id=9B3_DwAAQBAJ&oi=fnd&pg=PA21&dq=biological+and+chemical+pest+control+methods&ots=fq_tInOPHD&sig=yP-mk7_y3km7c3tgNC663ayHCGM&redir_esc=y#v=onepage&q=biological%20and%20chemical%20pest%20control%20methods&f=false

Pimentel, D., Acquay, H., Biltonen, M., Rice, P., Silva, M., Nelson, J., ... & D'amore, M. (1992). Environmental and economic costs of pesticide use. BioScience, 42(10), 750-760. https://doi.org/10.2307/1311994.

Blanket Application. PlantFacts. plantfacts.osu.edu/wiki/index.php/Blanket_application. Accessed 04 Aug. 2024.

Ali, M. A., Dhanaraj, R. K., & Kadry, S. (2024). AI-enabled IoT-based pest prevention and controlling system using sound analytics in large agricultural field. Computers and Electronics in Agriculture, 220, 108844. https://doi.org/10.1016/j.compag.2024.108844.

Cubero, S., Marco-Noales, E., Aleixos, N., Barbé, S., & Blasco, J. (2020). Robhortic: A field robot to detect pests and diseases in horticultural crops by proximal sensing. *Agriculture*, *10*(7), 276. https://doi.org/10.3390/agriculture10070276.