




Article

AgriNAS: Neural Architecture Search with Adaptive Convolution and Spatial–Time Augmentation Method for Soybean Diseases

Oluwatoyin Joy Omole ^{1,†}, Renata Lopes Rosa ^{1,†} , Muhammad Saadi ^{2,†} 
and Demóstenes Zegarra Rodriguez ^{1,*,†} 

¹ Department of Computer Science, Federal University of Lavras, Lavras 37200-000, Minas Gerais, Brazil; oluwatoyin.omole@estudante.ufla.br (O.J.O.); renata.rosa@ufla.br (R.L.R.)

² School of Science and Technology, Department of Computer Science, Nottingham Trent University, Nottingham NG11 8NS, UK; muhammad.saadi@ntu.ac.uk

* Correspondence: demostenes.zegarra@ufla.br

† These authors contributed equally to this work.

Abstract: Soybean is a critical agricultural commodity, serving as a vital source of protein and vegetable oil, and contributing significantly to the economies of producing nations. However, soybean yields are frequently compromised by disease and pest infestations, which, if not identified early, can lead to substantial production losses. To address this challenge, we propose AgriNAS, a method that integrates a Neural Architecture Search (NAS) framework with an adaptive convolutional architecture specifically designed for plant pathology. AgriNAS employs a novel data augmentation strategy and a Spatial–Time Augmentation (STA) method, and it utilizes a multi-stage convolutional network that dynamically adapts to the complexity of the input data. The proposed AgriNAS leverages powerful GPU resources to handle the intensive computational tasks involved in NAS and model training. The framework incorporates a bi-level optimization strategy and entropy-based regularization to enhance model robustness and prevent overfitting. AgriNAS achieves classification accuracies superior to VGG-19 and a transfer learning method using convolutional neural networks.

Keywords: neural architecture search; adaptive convolutional architecture; data augmentation; soybean disease and pest detection



Citation: Omole, O.J.; Rosa, R.L.; Saadi, M.; Rodriguez, D.Z. AgriNAS: Neural Architecture Search with Adaptive Convolution and Spatial–Time Augmentation Method for Soybean Diseases. *AI* **2024**, *5*, 2945–2966. <https://doi.org/10.3390/ai5040142>

Academic Editor: Arslan Munir

Received: 9 September 2024

Revised: 9 October 2024

Accepted: 22 October 2024

Published: 16 December 2024



Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Soybean is a globally significant legume crop, renowned for its high protein and oil content, making it a vital resource for both human and animal consumption [1]. Soybean is recognized as the plant that yields the highest protein per cultivated area, playing a crucial role in the global agricultural economy [2,3]. It accounts for 48% of the world's oil crop market, underlining its economic importance [4]. Soybean is extensively cultivated in countries such as the United States and Brazil, which together contribute over 80% of global exports [5]. The crop's significance is further highlighted by its utility in producing protein meal and vegetable oil, both of which are essential in the global food industry [6]. However, soybean yields are frequently threatened by pests and diseases, which can cause substantial losses [7]. Estimates suggest that pests alone are responsible for yield losses ranging between 26 and 29%, significantly impacting both the quantity and quality of soybean production [8,9]. Early and accurate detection of these threats is crucial for implementing timely interventions and minimizing yield losses [10].

Given the global importance of soybean as a staple crop, the ability to protect and enhance its yield through technological innovations is crucial. This has led to the increasing adoption of advanced artificial intelligence (AI) [11,12] techniques in agriculture, particularly in the areas of pest and disease detection. The convergence of agriculture

and AI represents a significant step forward in addressing the challenges of food security, resource optimization, and sustainable farming practices. Among these AI advancements, deep learning (DL) [13] has shown remarkable potential due to its capacity to analyze large datasets and identify complex patterns that are not easily discernible by traditional methods [14].

In recent years, deep learning models [15] have emerged as powerful tools for the detection of plant pests and diseases [16,17], offering significant advantages over traditional methods. Traditional techniques for detecting pests and diseases, such as manual inspection or simple rule-based image processing methods, are often labor-intensive, time-consuming, and prone to human error. Furthermore, these methods struggle to accurately detect subtle variations in symptoms or adapt to new disease manifestations without extensive reprogramming or manual intervention.

By contrast, deep learning models, specifically Convolutional Neural Networks (CNNs), excel in recognizing complex patterns in large datasets [18], making them particularly well-suited for tasks involving high variability and subtle symptoms in agricultural images. Quantitative comparisons have demonstrated that CNN-based methods typically achieve superior performance in classification tasks, with accuracy rates often exceeding 90% [19], significantly higher than the 70–80% accuracy commonly reported for traditional image processing techniques [20]. This is primarily due to CNNs' ability to automatically learn hierarchical features from data, ranging from low-level textures to high-level patterns, which traditional methods struggle to capture.

Moreover, deep learning models can generalize well across different datasets without the need for manual feature engineering, significantly reducing the time and expertise required for implementation [21,22]. For example, CNNs have been shown to reduce error rates by up to 25% compared to traditional methods in studies focused on plant disease detection [20]. This makes deep learning a far more scalable and adaptable approach for real-time agricultural applications, where new diseases or pest infestations can emerge rapidly.

However, as the complexity and size of datasets in agriculture continue to grow, the demand for more efficient and scalable deep learning models becomes paramount. Traditional methods of designing and training neural networks are often time-consuming and computationally expensive [23]. This is especially challenging in the agricultural domain, where data variability and environmental factors require models that are both robust and adaptable. The manual design of optimal CNN architectures for such tasks can be a daunting and error-prone process, often leading to suboptimal solutions that do not fully exploit the available data [24].

To address the challenges of accurately and efficiently detecting pests and diseases in agriculture, this work leverages the integration of Neural Architecture Search (NAS) frameworks, a promising solution for automating the design of high-performing neural networks tailored to specific tasks [25]. Traditional methods for neural network design often involve manual tuning and heuristic approaches, which can be time-consuming, error-prone, and suboptimal in performance. NAS systematically explores a vast search space of possible neural network configurations, identifying architectures that are specifically optimized for accuracy, efficiency, and robustness.

By automating the process of architecture design, NAS addresses key shortcomings in existing methods—namely, the reliance on manually designed models that may not generalize well across different agricultural environments or datasets. This work focuses on applying NAS to agricultural image analysis to develop neural networks that are better suited to handling the variability and complexity inherent in detecting plant pests and diseases. The proposed approach not only accelerates model development but also ensures that the resulting architectures are optimized for both performance and resource efficiency, which is crucial for real-world deployment in resource-constrained environments such as farms.

In agricultural contexts, the application of NAS offers several key advantages. Firstly, NAS can optimize models to handle the variability inherent in agricultural data, such as

differences in pest and disease manifestations across regions, variations in crop health due to environmental factors, and the presence of noise in data collected from the field. Secondly, NAS can be used to develop models that are computationally efficient, making them suitable for deployment in resource-constrained environments, such as rural areas where access to high-performance computing infrastructure may be limited. Finally, the use of NAS can lead to the discovery of novel architectures that outperform manually designed models, providing a competitive edge in terms of accuracy and robustness [26].

One of the most recent advancements in data augmentation, which plays a critical role in enhancing the robustness of deep learning models, is the introduction of Spatial–Time Augmentation (STA). This novel method, distinct from traditional Generative Adversarial Networks (GANs) used for data augmentation, incorporates both spatial and temporal variations to simulate real-world conditions in agricultural environments. While GANs have been extensively used to generate realistic synthetic data [27,28], STA offers a more dynamic approach by applying transformations inspired by relativity theory, enhancing the model’s ability to generalize across diverse scenarios. This method is particularly useful in agriculture, where temporal changes in pest and disease manifestations can significantly affect model accuracy. The STA method ensures that the augmented data not only increase the dataset size but also capture the variability seen in real-world applications, leading to superior model performance.

This paper introduces AgriNAS, a novel approach that combines NAS with adaptive convolutional architectures specifically designed for soybean pest and disease detection. AgriNAS leverages the power of NAS to automatically discover optimal CNN architectures that are tailored to the unique challenges of agricultural data. By integrating an advanced data augmentation strategy [29], AgriNAS enhances the detection of subtle disease features, improving the model’s ability to generalize across diverse agricultural conditions. Additionally, AgriNAS incorporates a multi-stage convolutional network [30] that dynamically adapts to the complexity of the input data, further optimizing both accuracy and computational efficiency.

The use of NAS in this context represents a major advancement in the field of precision agriculture, enabling the development of models that can more effectively address the challenges of pest and disease detection. Moreover, the flexibility of the AgriNAS framework allows it to be extended to other crops and agricultural challenges, highlighting its potential as a versatile tool for modern agriculture.

The contributions of this work include the following:

1. Development of AgriNAS, a tailored Neural Architecture Search framework with adaptive convolutional networks for enhanced detection of soybean pests and diseases.
2. Implementation of a novel data augmentation strategy, a Spatial–Time Augmentation (STA) method, improving the model’s ability to generalize across diverse agricultural conditions.
3. Exploration of the potential for expanding AgriNAS to other agricultural applications, showcasing its versatility and impact on the broader field of precision agriculture.

The motivation behind the use of STA lies in its ability to capture both spatial and temporal variability in agricultural data. While spatial transformations enhance the model’s ability to generalize across different visual perspectives, the inclusion of temporal augmentation simulates changes in pest and disease manifestations depending on factors such as time of day, season, and environmental changes. This temporal dimension is critical for detecting pests and diseases at different stages of development, making the model more robust in real-world agricultural environments where these factors evolve dynamically.

Our experimental results demonstrate that AgriNAS not only surpasses traditional CNNs but also outperforms other NAS-based solutions in terms of both accuracy and computational efficiency. The framework’s ability to adapt to different agricultural conditions and its scalability make it a promising solution for large-scale agricultural applications. Furthermore, the successful implementation of NAS in AgriNAS underscores the growing

importance of automated model development in AI, particularly in domains where data variability and complexity are significant challenges.

The remainder of this article is structured as follows: Section 2 reviews the existing literature on ML and DL methods for pest detection, focusing on advancements in NAS and its applications in agriculture. Section 3 outlines the methodology of AgriNAS, while Section 4 presents the results and discusses their implications. Finally, Section 5 concludes the paper and suggests potential directions for future research.

2. Related Works

This section provides an overview of key research areas relevant to this study, focusing on both the agricultural aspects of soybean pests and the computational algorithms employed for pest detection.

2.1. Soybean Caterpillars

Soybean crops are particularly vulnerable to a range of pests [31], with caterpillars being among the most damaging [32]. These pests can attack the soybean plant at various stages, leading to significant yield losses if not properly managed [33]. Research by [34] highlights the critical impact of bedbugs and caterpillars on soybean health. While soybeans can tolerate a certain level of defoliation—up to 33% during the vegetative stage—the tolerance decreases significantly during the pod set stage, where even 16% defoliation can result in notable yield losses [35]. Species such as the velvet bean caterpillar and the soybean looper are particularly aggressive, with the former capable of stripping an entire field within a week [36].

The dataset used in this study includes images of soybean plants affected by various caterpillar species, particularly those from the genus *Spodoptera* [37], which are known to cause significant damage during both the vegetative and reproductive stages of the plant [38]. The focus of this study is on the early detection of these pests to prevent irreversible damage.

In [39], the use of a modified Faster R-CNN model for detecting defoliation from canopy-level images was explored. However, the model's precision and recall were limited, emphasizing the need for more targeted approaches, such as using individual leaf images for better accuracy in detecting pest actions.

While recent advancements in Camouflaged Object Detection (COD) demonstrate potential for detecting objects that blend into their surroundings, applying these techniques directly to agricultural pest detection may not be ideal. One significant challenge is that COD methods, such as the Dual-Constraint Coarse-to-Fine Network [40], are typically designed for specific environments and may require extensive retraining to adapt to the unique complexities of agricultural settings. In agriculture, the variety of crops, pest types, and changing environmental conditions introduce considerable variability that COD techniques might not effectively accommodate. Furthermore, the computational complexity of COD methods could hinder real-time applications in resource-constrained environments, where swift detection is crucial for timely interventions.

2.2. Soybean Pests (*Diabrotica Speciosa*)

Diabrotica speciosa [38] is a significant pest in Latin America, particularly in soybean crops. As adults, these beetles act as defoliators, causing extensive damage to pods and flowers, and are particularly harmful to seedlings and young plants [41,42]. The study by [43] further confirmed that *Diabrotica speciosa* preferentially feeds on the tender leaves of soybean plants, making precise and early detection crucial for effective pest management.

Recent advances in the use of NAS for agricultural applications have shown significant promise in enhancing the detection and management of pests [44]. NAS frameworks automate the design of neural networks, optimizing their architectures for specific tasks such as image classification and object detection in agricultural settings. For example, Ref. [45] introduced DARTS, a method that efficiently explores the space of possible network

architectures, leading to optimized models that can be adapted for image detection tasks. In the context of image detection, studies have demonstrated that NAS-generated models outperform traditional CNN architectures by providing more accurate and computationally efficient solutions [46]. However, few studies have explored the use of DARTS in pest detection scenarios.

In [47], the NAS is used to develop a model specifically designed for detecting soybean diseases, showing significant improvements in accuracy compared to manually designed networks. This approach, while primarily focused on disease detection, highlights the potential of NAS to be adapted for broader pest management applications, including the detection of pests like *Diabrotica speciosa*. In [48], a multi-objective NAS framework is employed to balance the trade-offs between model complexity and detection accuracy in agricultural applications. The work emphasizes the adaptability of NAS frameworks to different agricultural environments and crop types, making it a promising approach to the early detection of pests such as *Diabrotica speciosa* in soybean crops.

The integration of NAS with advanced data augmentation techniques, as discussed in [49], further enhances the model's ability to generalize across varying environmental conditions, which is critical for detecting *Diabrotica speciosa* in diverse agricultural settings. These advancements underscore the growing role of NAS in developing robust, scalable, and efficient models for pest detection in agriculture. Although NAS offers optimized models for pest detection, it faces significant challenges such as high computational costs [50–52]. Additionally, practical implementation in agricultural environments requires advanced infrastructure and can be complex [53,54]. These limitations highlight the need for improvements to make NAS more accessible and robust [55].

2.3. Data Augmentation Techniques

Data augmentation has become a critical technique for improving the robustness and generalization of deep learning models, particularly in fields like agriculture where the variability in data can be immense. Traditional data augmentation techniques, such as random rotations, flips, scaling, and color adjustments, have been widely employed to artificially increase the size and diversity of training datasets, thus helping models generalize better to unseen data [56]. These techniques have proven effective in a variety of image-based tasks, including the detection of plant diseases and pests [57,58].

In the context of agricultural applications, where conditions and visual characteristics of crops can vary significantly due to environmental factors, data augmentation plays a pivotal role in ensuring that models perform reliably across different scenarios. For instance, Wang et al. [59] investigated the impact of various augmentation methods on the performance of Convolutional Neural Networks (CNNs) in plant disease detection. Their findings revealed that the implementation of comprehensive augmentation techniques, including random rotations, flips, and color adjustments, led to an improvement in model accuracy by over 15%, highlighting the necessity of augmenting training data to capture the variability inherent in agricultural settings.

More recently, advanced data augmentation methods have gained traction, notably the use of Generative Adversarial Networks (GANs) to generate synthetic training data [60]. GANs can produce highly realistic images that augment existing datasets, providing a robust solution to the limitations of traditional augmentation methods. For example, GAN-generated images have been shown to enhance the training set's diversity, resulting in performance improvements of up to 20% in CNNs tasked with disease identification. However, despite their potential, GANs introduce challenges, such as the risk of generating images that may not accurately represent the variations found in real agricultural data [61]. This discrepancy can lead to models that perform well on synthetic data but fail to generalize effectively to real-world conditions. Therefore, while GANs offer a promising avenue for data augmentation, careful consideration of their limitations is essential for practical agricultural applications.

In this study, the STA method is presented as an approach to data augmentation. Unlike traditional or GAN-based techniques, which often rely on static transformations such as rotations or flips, STA introduces both spatial and temporal variability by simulating the relative movement of objects, such as pests, within different time and space dimensions. This is achieved through the application of Lorentzian transformations to the spatial coordinates of the images, which alters the observational perspective dynamically. Additionally, STA introduces a noise component proportional to the simulated relative velocity of the objects, mimicking the inherent uncertainties and irregularities found in real-world agricultural environments.

The use of Lorentzian transformations ensures that the augmented images reflect realistic motion and changes in the position of objects over time, allowing the model to learn from more diverse and complex datasets. By incorporating temporal aspects, STA helps capture the evolving nature of pest infestations and disease symptoms, which is crucial for accurate detection under varying conditions. Furthermore, the addition of noise proportional to velocity enhances the robustness of the model by preparing it to handle data with potential observational inaccuracies.

2.4. Convolutional Neural Networks (CNNs) and Neural Architecture Search (NAS)

CNNs have become the standard for image classification tasks, particularly in agriculture, where precise identification of diseases and pests is critical [62]. CNNs are designed with multiple layers that progressively learn to detect various features, such as edges, shapes, and textures. This hierarchical learning makes them exceptionally well-suited for analyzing plant health from images [63,64].

Quantitatively, CNNs have shown remarkable performance improvements over traditional image classification methods. For instance, in comparison to manual feature extraction techniques that often yield accuracy rates below 80%, CNNs consistently achieve accuracies above 90% in various agricultural tasks. In a study by Bevers et al. [47], a CNN method was implemented, incorporating transfer learning, data engineering, and data augmentation, resulting in an impressive overall testing accuracy of 96.8%. This highlights the capacity of CNNs to learn complex patterns from large datasets, which is particularly beneficial in environments where subtle differences in plant health need to be identified.

However, while traditional CNN architectures have proven effective, they often require extensive computational resources [62], especially when dealing with large and complex datasets [65]. This necessitates the exploration of more efficient architectures through methods like Neural Architecture Search (NAS), which automates the design of neural networks. By optimizing architectures for specific tasks, NAS enables the development of CNNs that not only maintain high accuracy but also reduce computational costs, making them more practical for real-world agricultural applications.

To address these challenges, NAS has emerged as a powerful tool for automating the design of neural networks. NAS frameworks, such as DARTS (Differentiable Architecture Search), allow for the efficient exploration of different network architectures, optimizing them for specific tasks [25]. By automating the search process, NAS can identify architectures that are not only more accurate but also more computationally efficient.

In this study, AgriNAS is introduced, a method that integrates NAS with adaptive convolutional architectures specifically tailored for soybean pest and disease detection. AgriNAS leverages advanced data augmentation strategies to enhance the model's ability to detect subtle disease features, while dynamically adjusting the network architecture to handle varying complexities in the input data. This approach addresses the limitations of traditional CNNs by providing a scalable solution that is both robust and efficient.

3. Methodology

This section details the comprehensive steps involved in developing and validating the AgriNAS model, with in-depth explanations of each component and the software used for simulations.

Figure 1 presents the process, beginning with the collection of soybean leaf images, which then undergo preprocessing using Python 3.12 libraries such as OpenCV for size standardization. Following this, the Spatial–Time Augmentation (STA) method is applied, introducing both spatial and temporal variability into the images. This novel augmentation strategy simulates the relative movement of objects (e.g., pests) across different times and spaces by applying a Lorentzian transformation to the spatial coordinates of features, resulting in augmented images that reflect varying observational perspectives. Additionally, STA adds a noise factor proportional to the simulated relative velocity, ensuring that the augmented images are both realistic and diverse, providing a robust training set for the deep learning models. Subsequently, the NAS is employed within the AgriNAS framework using the DARTS algorithm, implemented in PyTorch. This search is guided by a bi-level optimization strategy, ensuring that the architecture is both effective and efficient. The selected architecture dynamically adjusts its convolutional layers based on the complexity of the input image, using TensorFlow 2.13.0 or PyTorch 2.2 to modify layer depth, convolution types, and filter sizes to capture relevant features.

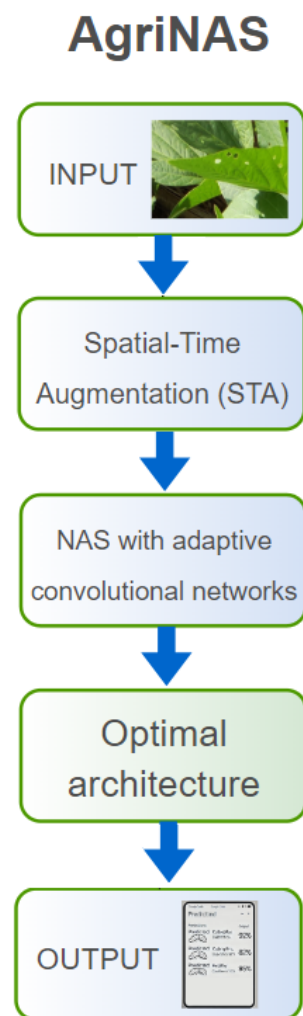


Figure 1. Steps involved in creating AgriNAS.

The AgriNAS framework is trained and optimized using Google Colab, which provides access to powerful NVIDIA Tesla K80 GPUs with 12GB of GPU memory. These advanced GPU resources significantly enhance the performance and efficiency of the system during the computationally intensive tasks associated with NAS and model training.

The parallel processing capabilities of the GPU accelerate the handling of large datasets and complex calculations, enabling real-time data augmentation and feature extraction.

Key parameters such as architecture depth, types of convolutions, and filter sizes are optimized by the NAS framework, and the use of powerful GPUs ensures faster model convergence, allowing the system to adapt and refine its architecture more quickly. Furthermore, this increased computational efficiency facilitates more extensive hyperparameter tuning and experimentation, contributing to improved model accuracy and robustness.

Initial testing and smaller-scale training were conducted on a local machine with an Intel Core i7 processor, 16 GB RAM, and an NVIDIA GeForce GTX 1060 GPU. The training procedure employed 5-fold cross-validation to evaluate the model's generalizability, with each model trained on 80% of the data and validated on the remaining 20%. Advanced regularization techniques, such as entropy-based regularization, were also employed to prevent overfitting and ensure the model's generalizability.

The trained model classifies the input images into categories such as Caterpillar, Diabrotica, and Healthy. The results are visualized using Matplotlib and Seaborn, providing clear outputs that can be used for monitoring and decision-making in agricultural practices.

3.1. Spatial–Time Augmentation (STA)

The Spatial–Time Augmentation (STA) method introduces a novel augmentation strategy that incorporates both spatial and temporal variability into agricultural images. Unlike traditional augmentation techniques that primarily focus on static transformations, STA simulates the relative movement of objects (e.g., pests) across different times and spaces. This is achieved by applying a Lorentzian transformation to the spatial coordinates of features, resulting in augmented images that reflect varying observational perspectives. Additionally, STA introduces a noise factor proportional to the simulated relative velocity, which accounts for observational uncertainties and enhances the realism of the augmented data. By generating a more diverse and realistic training set, STA contributes to improved model generalization and performance in deep learning tasks within agriculture.

Algorithm 1 describes the process of applying the Spatial–Time Augmentation (STA) technique to an original dataset D . The method begins by initializing key STA parameters: the velocity v , the Lorentz factor γ , and the noise factor ϵ .

Algorithm 1 AgriNAS: Neural Architecture Search Framework

- 1: **Input:** Initialize architecture parameters α , network weights w , learning rates η_α, η_w , and perturbation parameter ξ .
- 2: **while** not converged **do**
- 3: **Training Phase:** Update network weights w using gradient descent:

$$w \leftarrow w - \eta_w \nabla_w \mathcal{L}_{\text{train}}(w, \alpha)$$

- 4: **Search Phase:** Update architecture parameters α using second-order approximation:

$$\alpha \leftarrow \alpha - \eta_\alpha \nabla_\alpha \mathcal{L}_{\text{val}}(w - \xi \nabla_w \mathcal{L}_{\text{train}})$$

- 5: Apply regularization to architecture parameters to avoid overfitting:

$$\mathcal{L}_{\text{val}} \leftarrow \mathcal{L}_{\text{val}} + \lambda \sum_i H(\alpha_i)$$

- 6: **end while**

- 7: **Output:** Optimal architecture α^* and corresponding weights w^* .
-

1. **Spatial Transformation:** For each image x_i in the dataset, a spatial transformation such as translation or rotation is applied. This transformation simulates different observational angles or positions.

2. Lorentz Transformation: The spatial coordinates of each image are then transformed using a Lorentzian model:

$$x'_i = \gamma(x_i - vt)$$

$$t'_i = \gamma\left(t - \frac{vx_i}{c^2}\right)$$

Here, x'_i represents the transformed spatial coordinates, and t'_i represents the transformed temporal coordinates. These transformations account for the relative movement of objects in space and time.

3. Adding Noise: To simulate realistic environmental conditions, a Gaussian noise factor with a variance of 0.01 is added to the transformed image x'_i . This step introduces variability and uncertainty, effectively mimicking real-world scenarios where observations may be affected by noise or imprecision. By incorporating this level of noise, we enhance the model's robustness, ensuring it can generalize better across diverse agricultural contexts.
4. Augmented Dataset: The augmented image x'_i is then appended to the new dataset D' , which is used for training deep learning models.

Finally, the augmented dataset D' is returned, containing a diverse set of images that reflect different spatial and temporal scenarios, thus providing a robust training set for model development.

3.2. Dataset

The dataset used in this study, as described in [38], consists of images categorized into three folders: Caterpillar, Diabrotica, and Healthy. To enhance the dataset, additional images of healthy soybean leaves were sourced from Plant Village (<https://plantvillage.org/>, accessed on 12 March 2024), resulting in a total of 10,499 images. The final distribution of images across the categories is presented in Table 1.

Table 1. Number of images for Pest dataset.

Folder	Number of Images
Caterpillar	3309
Diabrotica Speciosa	2205
Healthy	4985
Total	10,499

The dataset used in this study consists of high-resolution images of soybean leaves, stems, and pods affected by various pests and diseases. These images were collected using a combination of high-definition field cameras. The images were captured under different environmental conditions, including varying light intensities and angles, to simulate real-world agricultural scenarios. This approach ensures diversity in the dataset, making the model more robust when detecting pests and diseases across different farming conditions. The dataset includes both healthy and infected samples, with careful attention to capturing early, middle, and advanced stages of pest and disease manifestation. Controlled lighting was used in some instances to improve image quality, while multi-angle shots were taken to ensure that the affected areas were adequately captured from different perspectives.

Additionally, the images were acquired using cameras with a minimum resolution of 12 megapixels in some instances, complementing the high-definition field cameras. This combination of imaging techniques not only enhances the dataset's resolution and detail but also allows for the capture of images in various field conditions. The use of cameras provides flexibility in accessing different areas of the fields while ensuring that high-quality images are obtained. Employing both high-definition cameras aims to create a comprehensive dataset that reflects the diversity and complexity of real agricultural environments, ultimately contributing to the model's ability to generalize effectively in

practical applications. The dataset is balanced, ensuring that both healthy and infected samples are adequately represented, with no class being over- or under-represented.

The preprocessing phase includes an advanced data augmentation technique known as Spatial–Time Augmentation (STA) to enhance the dataset’s variability and improve the model’s robustness. STA introduces both spatial and temporal variability into the images by simulating the relative movement of objects across different spatial and temporal contexts. This is achieved by applying a Lorentzian transformation to the spatial coordinates of features, resulting in augmented images that reflect different observational perspectives. Additionally, STA introduces a noise factor proportional to the simulated relative velocity, accounting for observational uncertainties and further enriching the diversity of the training set.

This augmentation process generates new training examples that simulate potential variations in real-world conditions, such as different lighting, angles, and damage patterns on the leaves. This process helps prevent overfitting and enables the model to generalize better to unseen data. The augmented dataset is then used as input for the NAS process, ensuring that the selected architecture is optimized across a broad range of scenarios.

Figure 2 illustrates healthy soybean leaves as captured in the dataset. In contrast, Figure 3 presents soybean leaves damaged by caterpillar infestation, while Figure 4 demonstrates the specific impact of the *Diabrotica Speciosa* pest on the soybean plants.



Figure 2. Healthy soybean leaves from the dataset.



Figure 3. Soybean leaves damaged by caterpillars.

The feature analysis was performed by examining color, texture, and shape features in the soybean leaves. Color histograms were used to compare the color distribution between healthy and infested leaves, indicating stages of pest infestation. Texture features were assessed using local binary patterns and a gray level co-occurrence matrix, focusing on differences in contrast and homogeneity. Shape features, including size and irregularity, were analyzed using Hu Moments to detect morphological changes due to pests. We also considered how environmental factors, such as lighting variations, could affect feature

extraction and model performance. Finally, class activation maps were employed to visualize the prioritized features during predictions, enhancing the interpretability of our results. This analysis provides valuable insights into the features contributing to the classification of soybean pest actions.



Figure 4. Soybean leaf damaged by *Diabrotica speciosa*.

3.3. Neural Architecture Search (NAS) Framework

The core of the proposed methodology is the AgriNAS framework, an advanced NAS framework specifically tailored for optimizing CNNs in the context of pest and disease detection in soybean crops. AgriNAS leverages an enhanced extension of DARTS to enable an efficient and comprehensive exploration of a vast architectural search space, designed to meet the unique challenges of agricultural image analysis.

AgriNAS's search space \mathcal{O} is meticulously crafted to include a diverse array of convolutional operations, such as standard convolutions, dilated convolutions, depthwise separable convolutions, pooling operations, and skip connections. Each operation $o_i \in \mathcal{O}$ is parameterized by learnable weights θ_{o_i} , allowing the framework to dynamically select and combine these operations to construct the optimal architecture for the specific task at hand.

The primary contribution of AgriNAS is its formulation of the architecture search as a bi-level optimization problem, which allows for a more structured and efficient approach to discovering optimal neural network configurations. In this framework, the goal is to minimize the validation loss \mathcal{L}_{val} with respect to the architecture parameters α , while simultaneously optimizing the network weights w to minimize the training loss $\mathcal{L}_{\text{train}}$. This dual optimization problem can be expressed mathematically as follows:

$$\min_{\alpha} \mathcal{L}_{\text{val}}(w^*(\alpha), \alpha), \quad (1)$$

subject to the constraint:

$$w^*(\alpha) = \arg \min_w \mathcal{L}_{\text{train}}(w, \alpha), \quad (2)$$

where $w^*(\alpha)$ denotes the optimal network weights for a given architecture α .

This bi-level optimization strategy effectively decouples the architecture search from the training process, enabling AgriNAS to focus on finding the best architecture while ensuring that the associated weights are optimized for performance. By allowing the model to adaptively select architectures that minimize validation loss, AgriNAS enhances its ability to generalize across diverse datasets.

Additionally, AgriNAS employs entropy-based regularization techniques to improve model robustness and prevent overfitting. This regularization is incorporated directly into the optimization process by penalizing architectures that exhibit excessive reliance on specific operations. The regularization term is defined as follows:

$$\mathcal{R}(\alpha) = \lambda \sum_i H(\alpha_i), \quad (3)$$

where $H(\alpha_i)$ represents the entropy of the operation probabilities associated with the architecture parameters, and λ is a hyperparameter controlling the strength of the regularization.

By promoting diversity in the selected operations and discouraging overly complex architectures, the entropy-based regularization enhances the model's ability to generalize to unseen data. This dual approach of bi-level optimization combined with regularization techniques ensures that AgriNAS not only identifies effective neural architectures but also maintains robustness against overfitting, ultimately resulting in a more reliable model for pest and disease detection.

To efficiently address this complex bi-level optimization, AgriNAS employs a gradient-based search algorithm enhanced by a second-order approximation. This approach mitigates the computational burden typically associated with exact second-order methods, ensuring that the architecture search remains computationally feasible even in large-scale agricultural datasets. The update rule for the architecture parameters is given by the following:

$$\alpha \leftarrow \alpha - \eta_\alpha \nabla_\alpha \mathcal{L}_{\text{val}}(w - \xi \nabla_w \mathcal{L}_{\text{train}}), \quad (4)$$

where η_α represents the learning rate for α , and ξ is a small perturbation parameter employed to approximate the second-order gradients. This update mechanism is a key contribution of AgriNAS, as it refines the architecture parameters in a manner that effectively balances model performance on the validation set with computational efficiency, considering the complex interactions between architecture and network weights.

Algorithm 2 outlines the implementation of the AgriNAS framework. The process involves several key steps, each crucial for optimizing the architecture and weights of a neural network designed for detecting soybean pests and diseases.

Algorithm 2 Spatial–Time Augmentation (STA)

- 1: **Input:** Original dataset $D = \{x_i, y_i\}_{i=1}^N$
- 2: **Output:** Augmented dataset D'
- 3: **Initialize:** STA parameters v, γ , noise factor ϵ
- 4: **for** each image $x_i \in D$ **do**
- 5: Apply spatial transformation to x_i (translation, rotation)
- 6: Compute Lorentz transformation on spatial coordinates:

$$x'_i = \gamma(x_i - vt)$$

$$t'_i = \gamma\left(t - \frac{vx_i}{c^2}\right)$$

- 7: Add relativistic noise to transformed image x'_i
 - 8: Append augmented image x'_i to D'
 - 9: **end for**
 - 10: **Return:** Augmented dataset D'
-

AgriNAS incorporates advanced regularization techniques to promote generalization and prevent overfitting. An entropy-based regularization term is applied directly to the architecture parameters α , expressed as follows:

$$\mathcal{R}(\alpha) = \lambda \sum_i H(\alpha_i), \quad \text{where} \quad H(\alpha_i) = - \sum_{o \in \mathcal{O}} \alpha_{io} \log \alpha_{io}. \quad (5)$$

This regularization term penalizes architectures that overly rely on a single operation, ensuring diversity in the selected operations and fostering robust performance across varying data distributions.

When employing NAS, the optimization process still requires careful tuning of key hyperparameters to ensure effective training and convergence of the selected architecture. The initial learning rate was set to 0.001, and a learning rate scheduler was employed to reduce the rate by a factor of 0.1 if no improvement in validation loss was observed over 5 epochs. This adjustment mechanism helped the NAS optimize architectures under varying learning conditions. A batch size of 32 was chosen, balancing memory constraints with computational efficiency, allowing the NAS to evaluate the performance of architectures on a sufficiently large subset of the data while maintaining manageable memory usage. The models were trained for up to 50 epochs, with early stopping based on validation loss to prevent overfitting, ensuring that the NAS-selected architectures were evaluated under conditions that allow for sufficient training without overfitting.

The Adam optimizer, known for its adaptive learning rate capabilities, was used, which was particularly beneficial for NAS as it allowed for the efficient training of diverse architectures with varying parameter sensitivities. Additionally, L2 regularization (weight decay) was applied with a factor of 0.01 to discourage overfitting, ensuring that the NAS framework prioritized architectures that generalize well. Dropout layers with a rate of 0.5 were incorporated to introduce robustness by randomly dropping units during training. These hyperparameters were integral to the NAS process, enabling the discovery of high-performing architectures tailored to the specific requirements of the dataset and task.

3.4. Adaptive Convolutional Architecture

The AgriNAS framework features an adaptive convolutional architecture that dynamically adjusts its complexity in response to the specific characteristics of the input data. This adaptability is vital for managing the variability inherent in agricultural image data, where manifestations of pests and diseases can differ significantly.

The core concept behind this adaptive architecture is to modulate the depth and configuration of the convolutional network based on the complexity of the input image x . Mathematically, the output of the convolutional network, parameterized by architecture parameters α and weights θ , is denoted as $f(x; \alpha, \theta)$. The network's adaptive behavior is governed by a function $\mathcal{F}(x)$, which determines the number of layers $L(x)$ and their respective configurations for each input image:

$$f(x; \alpha, \theta) = \sum_{i=1}^{L(x)} \mathcal{F}_i(x) \cdot f_i(x; \alpha_i, \theta_i), \quad (6)$$

where $L(x)$ represents the number of layers chosen by the adaptive function $\mathcal{F}(x)$, and $f_i(\cdot)$ corresponds to the operations performed in the i -th layer, parameterized by α_i and θ_i .

The adaptive function $\mathcal{F}(x)$ evaluates the complexity of the input image by assessing features such as texture, contrast, and the presence of multiple overlapping regions indicative of pest or disease severity. For more complex images, $\mathcal{F}(x)$ may increase the number of convolutional layers $L(x)$ and adjust filter sizes and pooling operations to capture more intricate patterns:

$$L(x) = \min(L_{\max}, \max(L_{\min}, \gamma \cdot \mathcal{C}(x))), \quad (7)$$

where L_{\max} and L_{\min} are predefined upper and lower bounds for the number of layers, γ is a scaling factor, and $\mathcal{C}(x)$ is a complexity measure derived from the input image. This complexity measure $\mathcal{C}(x)$ reflects a formal measure combining entropy, edge density, and pixel intensity variance of x , factors into a weighted function:

$$\mathcal{C}(x) = \lambda_1 H(x) + \lambda_2 D(x) + \lambda_3 V(x), \quad (8)$$

where $H(x)$ is the entropy, $D(x)$ is the edge density, and $V(x)$ is the pixel intensity variance, with $\lambda_1, \lambda_2, \lambda_3$ representing the respective weights for these components.

In addition to dynamically adjusting the number of layers, AgriNAS adapts the configuration of each layer, including the type of convolution (standard, dilated, or depthwise

separable), filter sizes, and activation functions. The adaptive function $\mathcal{F}_i(x)$ for each layer i is defined below.

The loss function $\mathcal{L}_{\text{train}}$ represents the training loss at layer i and is typically defined as a cross-entropy loss for classification tasks. The regularization term $\mathcal{R}(\alpha_i)$ penalizes overly complex architectures by introducing constraints on the architecture parameters α_i , preventing the model from becoming unnecessarily large or complex. This regularization is weighted by the hyperparameter λ , which controls the strength of the regularization, balancing between model complexity and performance.

$$\mathcal{F}_i(x) = \operatorname{argmin}_{\alpha_i, \theta_i} \{ \mathcal{L}_{\text{train}}(f_i(x; \alpha_i, \theta_i)) + \lambda \cdot \mathcal{R}(\alpha_i) \}, \quad (9)$$

where $\mathcal{L}_{\text{train}}$ measures the error between predicted and true labels, and $\mathcal{R}(\alpha_i)$ ensures the network remains computationally efficient and avoids overfitting.

Filter sizes are adaptively selected based on the complexity of the input image $C(x)$. Specifically, filter sizes increase with higher values of $C(x)$ to capture finer details in more complex images. This process involves dynamically scaling the filter size based on the computed complexity:

$$\text{Filter Size} = \alpha \cdot \text{Base Size} + \beta \cdot C(x) \quad (10)$$

where α and β are hyperparameters tuned during training.

It is important to note that the NAS framework explores various configurations of these parameters, selecting combinations that minimize training loss. Each network is trained with an optimized configuration tailored to the complexity of the input, ensuring a more efficient and adaptive model, rather than relying on predefined or static structures.

This adaptive architecture's optimization is integrated into the overall NAS framework, where both the architecture parameters α and the weights θ are updated iteratively. The gradient-based optimization of α and θ is performed using backpropagation through the dynamically adjusted network, ensuring the model converges to an architecture that is both effective for pest and disease detection and efficient in terms of computational resources.

The design of the multi-stage convolutional network incorporates various strategies for managing computational efficiency while ensuring effective feature extraction. Each layer is equipped with adaptive mechanisms that allow it to focus on relevant features based on the complexity of the input. For instance, in layers dealing with simpler images, the architecture might utilize fewer convolutional operations, while more complex images could trigger the use of multiple layers with varying filter types and sizes.

This multi-stage approach enables the model to dynamically allocate resources to layers that require more detailed processing, thereby optimizing the computational load. Additionally, the adaptive function $\mathcal{F}_i(x)$ not only determines the number of layers but also adapts the configuration of each layer, including the selection of activation functions and pooling strategies, ensuring that the network effectively captures intricate patterns indicative of pests and diseases. The flexibility inherent in this architecture allows AgriNAS to perform well across diverse agricultural conditions, improving both accuracy and efficiency in pest and disease detection tasks.

3.5. Performance Evaluation Metrics

The evaluation of the effectiveness of the AgriNAS algorithm is performed by measuring the results using several key performance metrics: accuracy, recall, precision, F-measure (F1 Score), and ROC AUC (Area Under the Curve).

1. Accuracy: Measures the ratio of correct predictions over the total number of instances evaluated. It is calculated as follows:

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \quad (11)$$

where True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN) values are extracted from the confusion matrix.

2. Recall: Also known as sensitivity, it measures the fraction of actual positive instances that were correctly identified by the model. It is crucial for evaluating performance, particularly in cases of imbalanced datasets. Recall is calculated as follows:

$$Recall = \frac{TP}{TP + FN} \quad (12)$$

3. Precision: Precision measures the proportion of positive identifications that were actually correct. It is calculated as follows:

$$Precision = \frac{TP}{TP + FP} \quad (13)$$

4. F-Measure (F1 Score): The F1 Score represents the harmonic mean of precision and recall, providing a single metric that balances the two. It is calculated as follows:

$$F\text{-measure} = \frac{2 \times (precision \times recall)}{precision + recall} \quad (14)$$

5. ROC Curve and AUC: The ROC curve plots the true positive rate (Recall) against the false positive rate, providing a graphical representation of a model's diagnostic ability. The Area Under the Curve (AUC) indicates the overall ranking performance, with a higher AUC representing a better model.

These metrics collectively offer a robust evaluation of the model's effectiveness in classifying soybean pest and disease categories. They are essential for understanding the strengths and limitations of the AgriNAS model compared to other models, such as VGG-19 and another model using CNN, in recognizing leaf diseases [47].

4. Results and Discussion

This section presents a comprehensive analysis of the results obtained from the AgriNAS framework in classifying soybean pest actions on leaves. The models evaluated include the AgriNAS-optimized CNN architecture, VGG-19, and another CNN model [47], a state-of-the-art NAS-based model from related work. The performance metrics used in this analysis are accuracy, recall, precision, F-measure, and ROC AUC, providing a thorough evaluation of each model's effectiveness.

In this study, we compared the performance of the AgriNAS-optimized CNN architecture with the VGG-19 and another CNN model from related work [47]. The AgriNAS-optimized CNN utilizes advanced features from Neural Architecture Search (NAS) to enhance pest classification. In contrast, VGG-19 is a well-established model with 19 layers, known for its effectiveness in image classification, employing small receptive fields and max pooling layers. The state-of-the-art CNN model referenced in [47] follows a standard convolutional architecture designed for agricultural images.

For training, we employed a learning rate of 0.001 and a batch size of 32, using the Adam optimizer for all models.

4.1. Performance Comparison

The evaluation of the models was conducted using the aforementioned performance metrics, with the results summarized in Table 2. To ensure that the observed differences in model performance are statistically significant, we performed a *t*-test comparing the results of AgriNAS with those of VGG-19 and another CNN model [47]. The *p*-values obtained confirm that AgriNAS significantly outperforms the other models across all metrics ($p < 0.05$).

Table 2. Test accuracies and evaluation metrics for the models.

Model	Acc.	Recall	Precis.	F-Meas.	AUC ROC
VGG-19	0.94	0.94	0.94	0.94	0.93
CNN model [47]	0.96	0.96	0.96	0.96	0.95
AgriNAS	0.98	0.98	0.99	0.98	0.97

4.2. Ablation Study

To further understand the contribution of each component within the AgriNAS framework, we conducted an ablation study by systematically removing or modifying specific elements, such as data augmentation and adaptive convolutional layers. The results, as shown in Table 3, highlight the significant impact of data augmentation and adaptive layers on the overall performance of the model. The removal of data augmentation led to a decrease in accuracy by approximately 3%, while disabling adaptive layers resulted in a 2% drop in F-measure.

Table 3. Ablation study results for AgriNAS.

Config.	Acc.	Rec.	Precis.	F-Meas.	AUC ROC
AgriNAS (Full)	0.98	0.98	0.99	0.98	0.97
No Augment.	0.95	0.96	0.97	0.96	0.95
No Adapt. Layers	0.96	0.97	0.97	0.96	0.96

4.3. Feature Map Visualization

To gain deeper insights into how the AgriNAS model interprets input images, from Figure 5, it is possible to visualize the feature maps generated by different convolutional layers. Figure 5 shows the activation patterns in various layers, highlighting how the model progressively abstracts and focuses on critical features like texture and leaf venation, which are indicative of specific diseases and pests. The feature maps demonstrate the model’s ability to capture intricate patterns that contribute to its high classification accuracy.

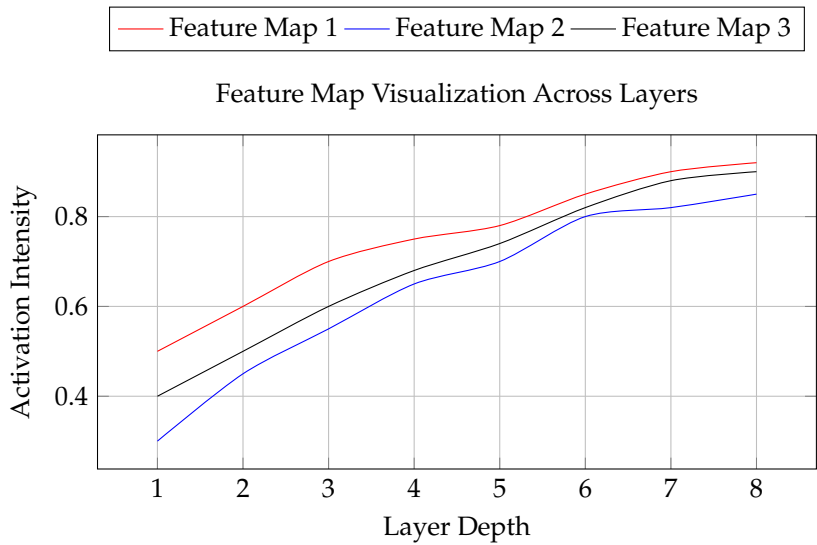


Figure 5. Visualization of feature maps from different layers of the AgriNAS model, showing how the model captures important features related to pest and disease detection.

4.4. Impact of Data Augmentation with STA

The data augmentation strategies employed in AgriNAS, with STA, play a crucial role in enhancing model robustness. Figure 6 illustrates the impact of data augmentation on model performance, comparing the accuracy, recall, and precision with and without STA. The results clearly show that data augmentation improved the model’s ability to generalize across different conditions, leading to a 3% increase in overall accuracy.

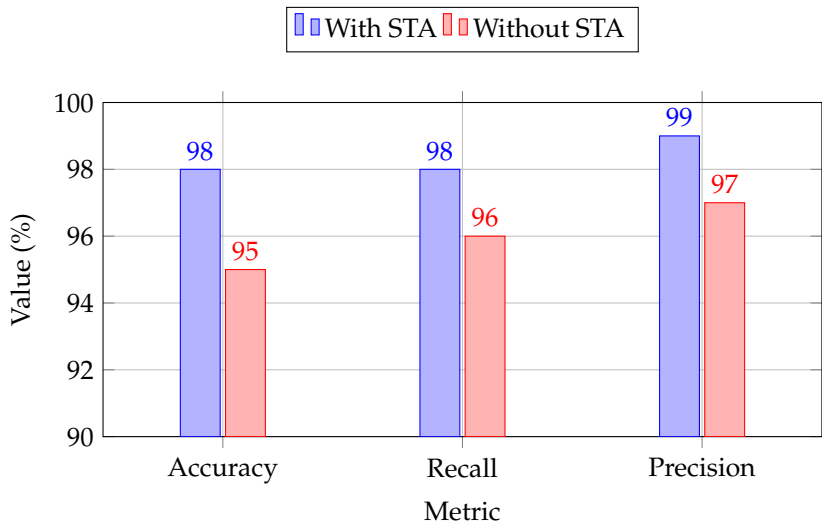


Figure 6. Impact of data augmentation on AgriNAS model performance. The results show significant improvements in accuracy, recall, and precision when data augmentation is applied.

4.5. Precision–Recall and ROC Curves

To further illustrate the performance differences among the models, Figure 7 shows the precision–recall curve for AgriNAS, another CNN model [47], and VGG-19. The adaptive convolutional layers in AgriNAS dynamically adjust based on input complexity, reducing unnecessary computational load. This, combined with effective regularization techniques and efficient search space exploration, leads to faster training times and lower GPU memory usage.

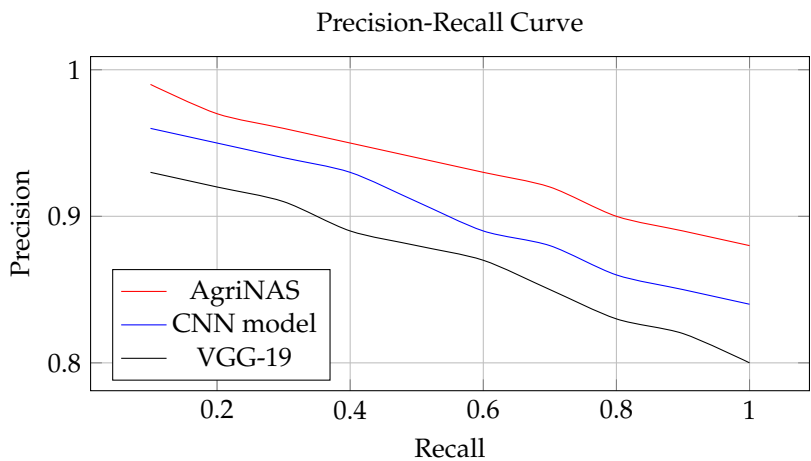


Figure 7. Precision–recall curve for AgriNAS, CNN model [47], and VGG-19.

4.6. Computation Time and Efficiency Analysis

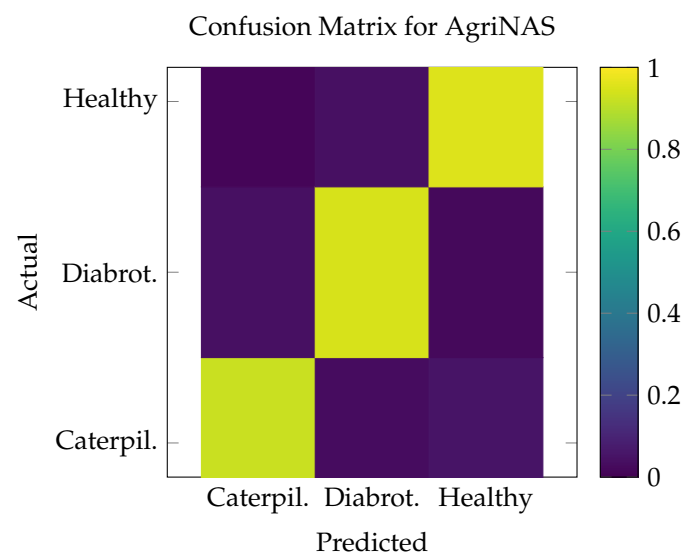
Table 4 presents the time and computational resources required to train the AgriNAS model compared to other models. This is particularly important for real-time applications in agriculture.

Table 4. Computation time and resource utilization.

Model	Training Time (hours)	GPU Memory (GB)
VGG-19	12	8
CNN model [47]	15	10
AgriNAS	10	7

4.7. Error Analysis

In Figure 8, the confusion matrix illustrates the classification performance of the AgriNAS model across different pest and disease categories: Caterpillar, Diabrotica, and Healthy. Each cell in the matrix represents the proportion of predictions made by the model for each actual category. The diagonal elements of the matrix (i.e., (1,1), (2,2), and (3,3)) show the correctly classified instances for each category. For example, the model correctly classified 92% of the Caterpillar images, 94% of the Diabrotica images, and 95% of the Healthy images. These high values indicate that the model has learned to accurately distinguish between the different categories. However, the off-diagonal elements indicate misclassification. For instance, 3% of the Caterpillar images were incorrectly classified as Diabrotica, and 5% were misclassified as Healthy. Similarly, 4% of the Healthy images were misclassified as Diabrotica. These misclassifications are relatively low, suggesting that the model is highly reliable but still has some difficulty differentiating between similar visual features.

**Figure 8.** Confusion matrix for AgriNAS, illustrating the classification performance across different pest and disease categories.

The performance of the AgriNAS algorithm can be attributed to several factors:

- **Adaptive Convolutional Architecture:** The AgriNAS model dynamically adjusts its network complexity based on the input data, allowing it to capture intricate patterns in the images that are indicative of specific pests and diseases. This adaptability results in better feature extraction and, consequently, higher classification accuracy.
- **Neural Architecture Search (NAS):** The use of NAS enables the automatic discovery of optimal network architectures tailored to the task of pest and disease detection. By exploring a wide range of possible architectures, AgriNAS identifies the most effective configuration, which contributes to its superior performance.
- **STA, Data Augmentation:** The model employs the STA technique, which introduces spatial and temporal variability into the dataset by simulating the relative movement of objects, such as pests, across different conditions. This approach enhances the model's ability to generalize by creating a diverse and realistic set of training images that

reflect various observational perspectives. The effectiveness of STA is demonstrated in the confusion matrix, where the model consistently maintains high accuracy across all categories, showcasing its robustness.

- **Regularization Techniques:** The integration of advanced regularization methods, including entropy-based regularization, prevents the model from overfitting to specific features. This ensures that the model remains effective across a wide range of scenarios, as evidenced by the balanced classification performance shown in the confusion matrix.

Overall, the AgriNAS algorithm's ability to accurately classify a diverse set of images while minimizing misclassification errors highlights its effectiveness in the context of soybean pest and disease detection.

5. Conclusions

The results of this study demonstrate the effectiveness of the AgriNAS framework in optimizing CNN architectures for the detection of soybean diseases. The use of a NAS framework allowed for the automated discovery of highly specialized network architectures tailored specifically to the nuances of soybean leaf imagery. By dynamically adjusting the complexity of the network through adaptive layer configuration, AgriNAS was able to capture the intricate patterns associated with different pest and disease manifestations. Moreover, the integration of the STA method played an important role in improving the model's generalization capabilities. STA introduced spatial and temporal variability into the dataset, simulating realistic scenarios that pests might encounter across different times and locations. This approach to data augmentation ensured that the model was trained on a diverse and realistic set of images, which, in turn, enhanced its ability to maintain high accuracy across all categories, as reflected in the confusion matrix. Additionally, the advanced regularization techniques within the NAS framework were critical to maintaining model robustness. The entropy-based regularization applied to the architecture parameters helped in preventing the model from becoming overly reliant on specific operations, thereby encouraging a diverse set of features to be learned. Furthermore, the second-order gradient-based optimization employed by AgriNAS enabled more precise updates to the architecture parameters, leading to faster convergence and a more efficient exploration of the search space. This method reduced the computational overhead typically associated with NAS, allowing AgriNAS to maintain computational efficiency without sacrificing performance. The scalability challenges faced by AgriNAS in real agricultural settings primarily revolve around high computational costs and the necessity for advanced infrastructure, particularly in resource-constrained environments such as rural areas. To address these issues, several strategies can be implemented. First, enhancing the computational efficiency of AgriNAS is essential, which can be achieved through techniques like model pruning, quantization, and the use of lightweight architectures that require fewer resources without sacrificing performance. Second, leveraging cloud computing can help mitigate the need for local high-performance infrastructure by providing scalable resources on-demand, allowing for intensive processing tasks to be performed remotely. Additionally, implementing edge computing solutions can further enhance scalability by processing data closer to the source, thus reducing latency and bandwidth requirements. Developing more efficient data collection and processing methods, such as transfer learning, can also assist in minimizing the amount of data needed for effective model training, which is particularly beneficial in regions with limited data availability. Finally, providing training and support for farmers and agricultural workers on utilizing AgriNAS effectively can empower users with the knowledge necessary for broader adoption and more effective implementation in diverse agricultural contexts. By implementing these strategies, AgriNAS can overcome scalability challenges and become a more practical solution for pest and disease detection in various agricultural settings, ensuring accessibility and effectiveness even in resource-limited environments. Future research should explore the extension of this framework to other crops and pests, potentially leveraging additional data modalities

such as multispectral or hyperspectral imagery. Additionally, the experimental evaluation of AgriNAS will be enhanced by incorporating more recent and advanced methods in the field of agricultural pest detection and deep learning. Specifically, we plan to include state-of-the-art techniques such as attention-based mechanisms, transformer networks, and more recent convolutional models. In future works, we will also expand our analysis to include a broader range of problematic cases affecting pest classification accuracy, such as overlapping visual features between different pest species, variations in pest life stages, and the impact of environmental factors like lighting conditions and occlusions on image quality. We also plan to explore the implications of these complexities on the confusion matrix results and consider incorporating more sophisticated evaluation metrics to capture model performance under challenging conditions. By addressing these shortcomings, we aim to enhance the robustness of our models and provide a more comprehensive framework for effectively classifying soybean pest actions in diverse agricultural environments.

Author Contributions: Conceptualization, O.J.O., R.L.R., M.S. and D.Z.R.; methodology, O.J.O., R.L.R., M.S. and D.Z.R.; software, O.J.O., R.L.R., M.S. and D.Z.R.; validation, O.J.O., R.L.R., M.S. and D.Z.R.; formal analysis, O.J.O., R.L.R., M.S. and D.Z.R.; investigation, O.J.O., R.L.R., M.S. and D.Z.R.; resources, O.J.O., R.L.R., M.S. and D.Z.R.; data curation, O.J.O., R.L.R., M.S. and D.Z.R.; writing—original draft preparation, O.J.O., R.L.R., M.S. and D.Z.R.; writing—review and editing, O.J.O., R.L.R., M.S. and D.Z.R.; visualization, O.J.O., R.L.R., M.S. and D.Z.R.; supervision, R.L.R. and D.Z.R.; project administration, D.Z.R.; funding acquisition, D.Z.R. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq) under Proc. No.: 404764/2021-5; and Fundação de Desenvolvimento Científico e Cultural (FINEP) under Proc. No. 2817/22.

Data Availability Statement: The data presented in this study are available on request from the corresponding author.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Hamza, M.; Basit, A.W.; Shehzadi, I.; Tufail, U.; Hassan, A.; Hussain, T.; Siddique, M.U.; Hayat, H.M. Global impact of soybean production: A review. *Asian J. Biochem. Genet. Mol. Biol.* **2024**, *16*, 12–20. [\[CrossRef\]](#)
2. Medic, J.; Atkinson, C.; Hurburgh, C.R. Current knowledge in soybean composition. *J. Am. Oil Chem. Soc.* **2014**, *91*, 363–384. [\[CrossRef\]](#)
3. Liu, K. *Soybeans: Chemistry, Technology, and Utilization*; Springer: Berlin/Heidelberg, Germany, 2012.
4. Xiaoming, Z.; Qiong, L. A Brief Introduction of Main Diseases and Insect Pests in Soybean Production in the Global Top Five Soybean Producing Countries. *Plant Dis. Pests* **2018**, *9*, 17.
5. Gale, F.; Valdes, C.; Ash, M. Interdependence of China, United States, and Brazil in soybean trade. In *US Department of Agriculture's Economic Research Service (ERS) Report*; US Department of Agriculture: New York, NY, USA, 2019; pp. 1–48.
6. Hartman, G.L.; West, E.D.; Herman, T.K. Crops that feed the World 2. Soybean—Worldwide production, use, and constraints caused by pathogens and pests. *Food Secur.* **2011**, *3*, 5–17. [\[CrossRef\]](#)
7. Nendel, C.; Reckling, M.; Debaeke, P.; Schulz, S.; Berg-Mohnicke, M.; Constantin, J.; Fronzek, S.; Hoffmann, M.; Jakšić, S.; Kersebaum, K.C.; et al. Future area expansion outweighs increasing drought risk for soybean in Europe. *Glob. Chang. Biol.* **2023**, *29*, 1340–1358. [\[CrossRef\]](#)
8. Oerke, E.C. Crop losses to pests. *J. Agric. Sci.* **2006**, *144*, 31–43. [\[CrossRef\]](#)
9. Rupe, J.; Luttrell, R.G. Effect of pests and diseases on soybean quality. In *Soybeans*; Elsevier: Amsterdam, The Netherlands, 2008; pp. 93–116.
10. Thanh Noi, P.; Kappas, M. Comparison of random forest, k-nearest neighbor, and support vector machine classifiers for land cover classification using Sentinel-2 imagery. *Sensors* **2017**, *18*, 18. [\[CrossRef\]](#)
11. Lokner Ladević, A.; Kramberger, T.; Kramberger, R.; Vlahek, D. Detection of AI-Generated Synthetic Images with a Lightweight CNN. *AI* **2024**, *5*, 1575–1593. [\[CrossRef\]](#)
12. Farea, A.; Yli-Harja, O.; Emmert-Streib, F. Understanding Physics-Informed Neural Networks: Techniques, Applications, Trends, and Challenges. *AI* **2024**, *5*, 1534–1557. [\[CrossRef\]](#)
13. Payán-Serrano, O.; Bojórquez, E.; Carrillo, J.; Bojórquez, J.; Leyva, H.; Rodríguez-Castellanos, A.; Carvajal, J.; Torres, J. Seismic Performance Prediction of RC, BRB and SDOF Structures Using Deep Learning and the Intensity Measure INp. *AI* **2024**, *5*, 1496–1516. [\[CrossRef\]](#)

14. Ribeiro, D.A.; Silva, J.C.; Lopes Rosa, R.; Saadi, M.; Mumtaz, S.; Wuttisittikulkij, L.; Zegarra Rodriguez, D.; Al Otaibi, S. Light field image quality enhancement by a lightweight deformable deep learning framework for intelligent transportation systems. *Electronics* **2021**, *10*, 1136. [\[CrossRef\]](#)
15. Silva, J.C.; Saadi, M.; Wuttisittikulkij, L.; Militani, D.R.; Rosa, R.L.; Rodriguez, D.Z.; Al Otaibi, S. Light-field imaging reconstruction using deep learning enabling intelligent autonomous transportation system. *IEEE Trans. Intell. Transp. Syst.* **2021**, *23*, 1587–1595. [\[CrossRef\]](#)
16. Shoaib, M.; Shah, B.; Ei-Sappagh, S.; Ali, A.; Ullah, A.; Alenezi, F.; Gechev, T.; Hussain, T.; Ali, F. An advanced deep learning models-based plant disease detection: A review of recent research. *Front. Plant Sci.* **2023**, *14*, 1158933.
17. Omole, O.J.; Rosa, R.L.; Rodriguez, D.Z. Soybean Disease Detection by Deep Learning Algorithms. In Proceedings of the 2023 International Conference on Software, Telecommunications and Computer Networks (SoftCOM), Split, Croatia, 21–23 September 2023; pp. 1–5.
18. Ferentinos, K.P. Deep learning models for plant disease detection and diagnosis. *Comput. Electron. Agric.* **2018**, *145*, 311–318. [\[CrossRef\]](#)
19. Tammina, M.R.; Sumana, K.; Singh, P.P.; Lakshmi, T.V.; Pande, S.D. Prediction of Plant Disease Using Artificial Intelligence. In *Microbial Data Intelligence and Computational Techniques for Sustainable Computing*; Springer: Berlin/Heidelberg, Germany, 2024; pp. 25–48.
20. Ghazal, S.; Kommineni, N.; Munir, A. Comparative Analysis of Machine Learning Techniques Using RGB Imaging for Nitrogen Stress Detection in Maize. *AI* **2024**, *5*, 1286–1300. [\[CrossRef\]](#)
21. Shruthi, U.; Nagaveni, V.; Raghavendra, B. A review on machine learning classification techniques for plant disease detection. In Proceedings of the 2019 5th International Conference on Advanced Computing & Communication Systems (ICACCS), Coimbatore, India, 15–16 March 2019; pp. 281–284.
22. Janiesch, C.; Zschech, P.; Heinrich, K. Machine learning and deep learning. *Electron. Mark.* **2021**, *31*, 685–695. [\[CrossRef\]](#)
23. Mohaidat, T.; Khalil, K. A survey on neural network hardware accelerators. *IEEE Trans. Artif. Intell.* **2024**, *5*, 3801–3822. [\[CrossRef\]](#)
24. Malekloo, A.; Ozer, E.; AlHamaydeh, M.; Girolami, M. Machine learning and structural health monitoring overview with emerging technology and high-dimensional data source highlights. *Struct. Health Monit.* **2022**, *21*, 1906–1955. [\[CrossRef\]](#)
25. Elsken, T.; Metzen, J.H.; Hutter, F. Neural architecture search: A survey. *J. Mach. Learn. Res.* **2019**, *20*, 1–21.
26. Chitty-Venkata, K.T.; Emani, M.; Vishwanath, V.; Somani, A.K. Neural architecture search benchmarks: Insights and survey. *IEEE Access* **2023**, *11*, 25217–25236. [\[CrossRef\]](#)
27. Goodfellow, I.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; Bengio, Y. Generative adversarial nets. *Adv. Neural Inf. Process. Syst.* **2014**, *27*.
28. Wang, X.; Yu, K.; Wu, S.; Gu, J.; Liu, Y.; Dong, C.; Qiao, Y.; Change Loy, C. Esrgan: Enhanced super-resolution generative adversarial networks. In Proceedings of the European Conference on Computer Vision (ECCV) Workshops, Munich, Germany, 8–14 September 2018.
29. Naveed, H.; Anwar, S.; Hayat, M.; Javed, K.; Mian, A. Survey: Image mixing and deleting for data augmentation. *Eng. Appl. Artif. Intell.* **2024**, *131*, 107791. [\[CrossRef\]](#)
30. Lu, M.; Jiao, S.; Deng, J.; Wang, C.; Zhang, Z. Efficient model updating of shaft-raft-hull system using multi-stage convolutional neural network combined with sensitivity analysis. *Ocean. Eng.* **2024**, *312*, 119041. [\[CrossRef\]](#)
31. Kheam, S.; Rubene, D.; Markovic, D.; Ith, S.; Uk, O.N.; Soung, S.; Ninkovic, V. The effects of cultivar mixtures on insect pest and natural enemy abundance, diseases, and yield in tropical soybean cropping system. *Biol. Control* **2024**, *196*, 105571. [\[CrossRef\]](#)
32. Romero, B.; Dillon, F.M.; Zavala, J.A. Different soybean cultivars respond differentially to damage in a herbivore-specific manner and decrease herbivore performance. *Arthropod-Plant Interact.* **2020**, *14*, 89–99. [\[CrossRef\]](#)
33. Yadav, V.; Khare, C.; Tiwari, P.; Srivastava, J. Important diseases of soybean crop and their management. In *Diseases of Field Crops Diagnosis and Management*; Apple Academic Press: Palm Bay, FL, USA, 2020; pp. 145–171.
34. Grande, M.L.M.; Rando, J.S.S. Integrated pest control adopted by soybean and corn farmers in Londrina, Paraná state, Brazil. *Arq. Inst. Biológico* **2018**, *85*, e0242015. [\[CrossRef\]](#)
35. GRDC, Grains Research and Development Corporation. *Soybean Section 7 Insect Control*; GRDC, Grains Research and Development Corporation: Barton, Australia, 2016.
36. Hodgson, E.W.; Koch, R.L.; Davis, J.A.; Reisig, D.; Paula-Moraes, S.V. Identification and biology of common caterpillars in us soybean. *J. Integr. Pest Manag.* **2021**, *12*, 13. [\[CrossRef\]](#)
37. Justus, C.M.; Paula-Moraes, S.V.; Pasini, A.; Hoback, W.W.; Hayashida, R.; de Freitas Bueno, A. Simulated soybean pod and flower injuries and economic thresholds for *Spodoptera eridania* (Lepidoptera: Noctuidae) management decisions. *Crop Prot.* **2022**, *155*, 105936. [\[CrossRef\]](#)
38. Mignoni, M.E.; Honorato, A.; Kunst, R.; Righi, R.; Massuquetti, A. Soybean images dataset for caterpillar and *Diabrotica speciosa* pest detection and classification. *Data Brief* **2022**, *40*, 107756. [\[CrossRef\]](#)
39. Walker, H. Detection of Insect-Induced Defoliation in Soybeans with Deep Learning and Object Detection. Ph.D. Thesis, Kansas State University, Manhattan, KS, USA, 2021.
40. Yue, G.; Xiao, H.; Xie, H.; Zhou, T.; Zhou, W.; Yan, W.; Zhao, B.; Wang, T.; Jiang, Q. Dual-constraint coarse-to-fine network for camouflaged object detection. *IEEE Trans. Circuits Syst. Video Technol.* **2023**, *5*, 3286–3298. [\[CrossRef\]](#)

41. Cabrera Walsh, G.; Ávila, C.J.; Cabrera, N.; Nava, D.E.; de Sene Pinto, A.; Weber, D.C. Biology and management of pest *Diabrotica* species in South America. *Insects* **2020**, *11*, 421. [\[CrossRef\]](#)
42. Costa, E.N.; Nogueira, L.; De Souza, B.H.S.; Ribeiro, Z.A.; Louvandini, H.; Zukoff, S.N.; Júnior, A.L.B. Characterization of antibiosis to *Diabrotica speciosa* (Coleoptera: Chrysomelidae) in Brazilian maize landraces. *J. Econ. Entomol.* **2018**, *111*, 454–462. [\[CrossRef\]](#)
43. Ávila, C.J.; Bitencourt, D.R.; da Silva, I.F. Biology, Reproductive Capacity, and Foliar Consumption of *Diabrotica Speciosa* (Germar) (Coleoptera: Chrysomelidae) in Different Host Plants. *Embrapa Agropecuária-Oeste-Artig. Periódico Indexado (ALICE)*. 2019. Available online: <https://ccsenet.org/journal/index.php/jas/article/view/0/39111> (accessed on 8 September 2024).
44. Ye, Y.; Chen, Y.; Xiong, S. Field detection of pests based on adaptive feature fusion and evolutionary neural architecture search. *Comput. Electron. Agric.* **2024**, *221*, 108936. [\[CrossRef\]](#)
45. He, H.; Liu, L.; Zhang, H.; Zheng, N. IS-DARTS: Stabilizing DARTS through Precise Measurement on Candidate Importance. In Proceedings of the AAAI Conference on Artificial Intelligence, Washington, DC, USA, 20–27 February 2024; Volume 38, pp. 12367–12375.
46. Weng, Y.; Zhou, T.; Liu, L.; Xia, C. Automatic Convolutional Neural Architecture Search for Image Classification Under Different Scenes. *IEEE Access* **2019**, *7*, 38495–38506. [\[CrossRef\]](#)
47. Bevers, N.; Sikora, E.J.; Hardy, N.B. Soybean disease identification using original field images and transfer learning with convolutional neural networks. *Comput. Electron. Agric.* **2022**, *203*, 107449. [\[CrossRef\]](#)
48. Ma, A.; Wan, Y.; Zhong, Y.; Wang, J.; Zhang, L. SceneNet: Remote sensing scene classification deep learning network using multi-objective neural evolution architecture search. *ISPRS J. Photogramm. Remote Sens.* **2021**, *172*, 171–188. [\[CrossRef\]](#)
49. Tiwari, R.G.; Maheshwari, H.; Agarwal, A.K.; Jain, V. HECNNNet: Hybrid Ensemble Convolutional Neural Network Model with Multi-Backbone Feature Extractors for Soybean Disease Classification. In Proceedings of the 2024 2nd International Conference on Intelligent Data Communication Technologies and Internet of Things (IDCIoT), Bengaluru, India, 4–6 January 2024; pp. 813–818.
50. Slimani, H.; El Mhamdi, J.; Jilbab, A. Optimizing Multi-Level Crop Disease Identification using Advanced Neural Architecture Search in Deep Transfer Learning. *Int. J. Comput. Digit. Syst.* **2024**, *16*, 1293–1306. [\[CrossRef\]](#)
51. Dewi, C.; Thiruvady, D.; Zaidi, N. Fruit Classification System with Deep Learning and Neural Architecture Search. *arXiv* **2024**, arXiv:2406.01869.
52. Chen, Y.; Liang, H.; Jiao, S. NAS-MFF: NAS-guided multiscale feature fusion network with pareto optimization for sonar images classification. *IEEE Sensors J.* **2024**, *24*, 14656–14667. [\[CrossRef\]](#)
53. Liu, Y.; Sun, Y.; Xue, B.; Zhang, M.; Yen, G.G.; Tan, K.C. A survey on evolutionary neural architecture search. *IEEE Trans. Neural Netw. Learn. Syst.* **2021**, *34*, 550–570. [\[CrossRef\]](#)
54. Gudzius, P.; Kurasova, O.; Darulis, V.; Filatovas, E. AutoML-based neural architecture search for object recognition in satellite imagery. *Remote Sens.* **2022**, *15*, 91. [\[CrossRef\]](#)
55. Ren, P.; Xiao, Y.; Chang, X.; Huang, P.Y.; Li, Z.; Chen, X.; Wang, X. A comprehensive survey of neural architecture search: Challenges and solutions. *ACM Comput. Surv.s (CSUR)* **2021**, *54*, 1–34. [\[CrossRef\]](#)
56. Maharana, K.; Mondal, S.; Nemade, B. A review: Data pre-processing and data augmentation techniques. *Glob. Transitions Proc.* **2022**, *3*, 91–99. [\[CrossRef\]](#)
57. Saleem, M.H.; Khanchi, S.; Potgieter, J.; Arif, K.M. Image-based plant disease identification by deep learning meta-architectures. *Plants* **2020**, *9*, 1451. [\[CrossRef\]](#)
58. Krishnaswamy Rangarajan, A.; Purushothaman, R. Disease classification in eggplant using pre-trained VGG16 and MSVM. *Sci. Rep.* **2020**, *10*, 2322. [\[CrossRef\]](#)
59. Wang, J.; Perez, L. The effectiveness of data augmentation in image classification using deep learning. *Convolutional Neural Netw. Vis. Recognit.* **2017**, *11*, 1–8.
60. Frid-Adar, M.; Diamant, I.; Klang, E.; Amitai, M.; Goldberger, J.; Greenspan, H. GAN-based synthetic medical image augmentation for increased CNN performance in liver lesion classification. *Neurocomputing* **2018**, *321*, 321–331. [\[CrossRef\]](#)
61. Makhoulouf, A.; Maayah, M.; Abughanam, N.; Catal, C. The use of generative adversarial networks in medical image augmentation. *Neural Comput. Appl.* **2023**, *35*, 24055–24068. [\[CrossRef\]](#)
62. Alzubaidi, L.; Zhang, J.; Humaidi, A.J.; Al-Dujaili, A.; Duan, Y.; Al-Shamma, O.; Santamaría, J.; Fadhel, M.A.; Al-Amidie, M.; Farhan, L. Review of deep learning: Concepts, CNN architectures, challenges, applications, future directions. *J. Big Data* **2021**, *8*, 1–74. [\[CrossRef\]](#)
63. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. Imagenet classification with deep convolutional neural networks. *Commun. ACM* **2017**, *60*, 84–90. [\[CrossRef\]](#)
64. Simonyan, K.; Zisserman, A. Very deep convolutional networks for large-scale image recognition. *arXiv* **2014**, arXiv:1409.1556.
65. Salehi, A.W.; Khan, S.; Gupta, G.; Alabdullah, B.I.; Almjally, A.; Alsolai, H.; Siddiqui, T.; Mellit, A. A study of CNN and transfer learning in medical imaging: Advantages, challenges, future scope. *Sustainability* **2023**, *15*, 5930. [\[CrossRef\]](#)

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.