








Article

Deep Learning-Based Classification of Macrofungi: Comparative Analysis of Advanced Models for Accurate Fungi Identification

Sifa Ozsari ¹, Eda Kumru ², Fatih Ekinci ³, Ilgaz Akata ⁴, Mehmet Serdar Guzel ¹, Koray Acici ⁵,
Eray Ozcan ¹ and Tunc Asuroglu ^{6,7,*}

¹ Department of Computer Engineering, Faculty of Engineering, Ankara University, Ankara 06830, Türkiye; ozsaris@ankara.edu.tr (S.O.); mguzel@ankara.edu.tr (M.S.G.); eozcan@teus.com.tr (E.O.)

² Graduate School of Natural and Applied Sciences, Ankara University, Ankara 06830, Türkiye; edakumru98@gmail.com

³ Department of Medical Physics, Institute of Nuclear Sciences, Ankara University, Ankara 06100, Türkiye; fatihkekinci@ankara.edu.tr

⁴ Department of Biology, Faculty of Science, Ankara University, Ankara 06100, Türkiye; akata@science.ankara.edu.tr

⁵ Department of Artificial Intelligence and Data Engineering, Faculty of Engineering, Ankara University, Ankara 06830, Türkiye; kacici@ankara.edu.tr

⁶ Faculty of Medicine and Health Technology, Tampere University, 33720 Tampere, Finland

⁷ VTT Technical Research Centre of Finland, 33101 Tampere, Finland

* Correspondence: tunc.asuroglu@tuni.fi

Abstract: This study focuses on the classification of six different macrofungi species using advanced deep learning techniques. Fungi species, such as *Amanita pantherina*, *Boletus edulis*, *Cantharellus cibarius*, *Lactarius deliciosus*, *Pleurotus ostreatus* and *Tricholoma terreum* were chosen based on their ecological importance and distinct morphological characteristics. The research employed 5 different machine learning techniques and 12 deep learning models, including DenseNet121, MobileNetV2, ConvNeXt, EfficientNet, and swin transformers, to evaluate their performance in identifying fungi from images. The DenseNet121 model demonstrated the highest accuracy (92%) and AUC score (95%), making it the most effective in distinguishing between species. The study also revealed that transformer-based models, particularly the swin transformer, were less effective, suggesting room for improvement in their application to this task. Further advancements in macrofungi classification could be achieved by expanding datasets, incorporating additional data types such as biochemical, electron microscopy, and RNA/DNA sequences, and using ensemble methods to enhance model performance. The findings contribute valuable insights into both the use of deep learning for biodiversity research and the ecological conservation of macrofungi species.

Keywords: macrofungi classification; deep learning; DenseNet121; fungi identification; machine learning models



Citation: Ozsari, S.; Kumru, E.; Ekinci, F.; Akata, I.; Guzel, M.S.; Acici, K.; Ozcan, E.; Asuroglu, T. Deep Learning-Based Classification of Macrofungi: Comparative Analysis of Advanced Models for Accurate Fungi Identification. *Sensors* **2024**, *24*, 7189. <https://doi.org/10.3390/s24227189>

Academic Editors: Zhaoyang Wang, Minh P. Vo, Hieu Nguyen and John Hyatt

Received: 9 October 2024

Revised: 7 November 2024

Accepted: 8 November 2024

Published: 9 November 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

Macrofungi, a group of fungi producing prominent fruiting bodies above or below the ground, represent an immense diversity within the species Basidiomycota and Ascomycota under the kingdom Fungi. These fruiting bodies, commonly known as mushrooms, are easily identifiable in natural environments and are vital components of forest ecosystems [1]. Macrofungi sustain ecosystem functionality by contributing to nutrient cycling, soil formation, and complex biotic interactions with other organisms [2]. To date, over 56,000 species of macrofungi have been documented worldwide, highlighting their remarkable biodiversity and ecological significance. Notably, approximately 1000 species are considered toxic, while more than 2000 species are edible, providing nutritional value and supporting food security for both humans and forest fauna [2–5].

Among the diverse macrofungi, many species exhibit unique ecological relationships and have significant cultural, nutritional, or toxicological relevance. For instance, *Amanita pantherina*, also known as the panther cap, is associated with coniferous and deciduous woodlands and has a storied history in mystical practices due to its psychoactive components, ibotenic acid and muscimol [6,7]. In contrast, *Boletus edulis*, or porcini, is one of the most popular wild edible mushrooms, valued for its rich nutritional content, including proteins, vitamins, and essential minerals [8]. Similarly, *Cantharellus cibarius*, the chanterelle, is cherished not only for its flavor and culinary value but also for its significant levels of bioavailable vitamin D2, crucial for human health [9].

Other notable macrofungi include *Lactarius deliciosus*, or the delicious milk cap, which plays an essential ecological role as an ectomycorrhizal partner in pine forests, promoting their growth and health [10]. Likewise, *Pleurotus ostreatus*, commonly known as the oyster mushroom, is not only a commercially important edible species but also contributes to environmental sustainability through the bioconversion of agricultural waste [11,12]. In contrast, *Tricholoma terreum*, previously considered a desirable edible species, has been re-evaluated due to potential health risks linked to rhabdomyolysis, underscoring the complex interplay between macrofungi and human health [13].

The advancement of machine learning (ML) and computer vision technologies is revolutionizing macrofungi identification and classification, offering significant advantages over traditional taxonomic methods. Conventional identification requires extensive expertise and can be time-consuming, whereas ML-based systems can rapidly analyze large datasets, discerning even subtle morphological variations with high precision [14,15]. Such technological integration accelerates species identification and enhances accessibility for non-experts through user-friendly mobile applications, thereby reducing misidentification risks and promoting broader public engagement in fungal biodiversity [16]. Furthermore, incorporating environmental data into ML models enhances ecological studies, contributing to the conservation of macrofungal species [17]. The synergy between researchers and citizen scientists has proven instrumental in creating high-quality datasets essential for effective conservation strategies.

The current research aims to leverage deep learning (DL), an advanced subset of artificial intelligence, to classify six distinct macrofungi species, namely *Amanita pantherina*, *Boletus edulis*, *Cantharellus cibarius*, *Lactarius deliciosus*, *Pleurotus ostreatus*, and *Tricholoma terreum*. These species were selected based on their ecological significance and diverse morphological characteristics. The development of a DL model for the rapid and accurate identification of these species will support scientific research and enhance public awareness and engagement, ultimately contributing to the conservation and sustainable use of macrofungal biodiversity.

2. Materials and Methods

In this study, datasets were generated using naturally captured images, subsequently labeled for the training of artificial intelligence (AI) algorithms (Figure 1). Data deemed unsuitable for training purposes were excluded from the final dataset. The images were then converted into the PNG format, with a resolution of 300 dpi to ensure consistency across the dataset. Given the requirement for a large and suitable dataset for AI algorithm training, additional data were sourced from open access repositories to supplement the natural data. Specifically, open-source data (www.gbif.org, accessed on 8 October 2024) accounted for no more than 25% of the total data for each mushroom species. Detailed information regarding the sources and characteristics of the datasets derived from these open sources is presented in Table 1. The primary objective was to train AI algorithms; thus, the images were not presented in their raw form within the article. Considering the substantial data requirements for effective AI training, it became evident that obtaining a dataset of the required size solely from natural shots would be impractical. Therefore, publicly available datasets were utilized to bridge this gap and meet the necessary data volume for robust AI training.



Figure 1. Overview of datasets utilized for training AI algorithms, presented from a macroscopic perspective.

Table 1. Type names of datasets taken from open sources, source name, percentage share of the taken dataset in the entire used dataset, and the technical specifications of the sources.

Mushroom Species Name	Source	Percentage of Photographs Taken from Source	Photo Type	Resolution
<i>Amanita pantherina</i>	Global Core Biodata Resource	<25%	JPG	300 dpi
<i>Boletus edulis</i>	Global Core Biodata Resource	<20%	JPG	300 dpi
<i>Cantharellus cibarius</i>	Global Core Biodata Resource	<25%	JPG	300 dpi
<i>Lactarius deliciosus</i>	Global Core Biodata Resource	<25%	JPG	300 dpi
<i>Pleurotus ostreatus</i>	Global Core Biodata Resource	<25%	JPG	300 dpi
<i>Tricholoma terreum</i>	Global Core Biodata Resource	<25%	JPG	300 dpi

Deep learning has evolved significantly since its early foundations in the 1940s. The concept of artificial neurons, introduced by Warren McCulloch and Walter Pitts [18], laid the groundwork for neural networks, followed by Frank Rosenblatt's perceptron in 1957 [19]. However, neural networks faced limitations, particularly with non-linear problems, which led to a decline in interest after criticisms in the late 1960s. The field resurged in the 1980s with the reintroduction of backpropagation by Geoffrey Hinton and colleagues [20], allowing multi-layer networks to be trained more effectively. Despite early progress, deep learning faced computational and data limitations until the 2000s, when breakthroughs like Hinton's deep belief networks and Alex Krizhevsky's convolutional neural networks (CNNs) [21] in the 2012 ImageNet competition demonstrated the power of deep architectures. The availability of GPUs and large datasets accelerated these advancements. The late 2010s saw further breakthroughs with the development of generative models like GANs and transformers, which revolutionized fields like image generation and natural language processing. Today, deep learning thrives due to improved algorithms, big data, and computational power, driving innovation across industries from healthcare to robotics.

In this study, 5 different machine learning techniques, logistic regression, Support Vector Machine (SVM), k-Nearest Neighbors (k-NN), decision tree, and random forest algorithms, and 12 different deep learning methods, ConvNeXtBase [22], ConvNeXtSmall [22], ConvNeXt-Large [22], EfficientNetB [23], EfficientNetB3 [23], EfficientNetB7 [23], DenseNet121 [24], InceptionV3 [25], InceptionResNetV2 [26], MobileNetV2 [27], ResNet152 [28], Xception [29], and swin transformers (shifted window transformers) [30] were used for the classification of six different fungi species.

2.1. ML

Logistic regression [31] is a linear model used for binary classification tasks. It estimates the probability that a given input belongs to a specific class by applying the logistic function to a linear combination of the input features. The output is a probability score between 0 and 1, which is then used to classify the instance. It works best when the classes are linearly separable and is easy to implement and interpret. However, it may struggle with more complex relationships in the data.

SVM [32] is a powerful classification algorithm that seeks to find the optimal hyperplane that best separates the classes in the feature space. It works by maximizing the margin between data points from different classes. SVM is effective in high-dimensional spaces and with non-linearly separable data when using kernel tricks. Despite its power, SVM can be sensitive to noisy data and may require the careful tuning of parameters (such as the choice of kernel) to achieve optimal performance.

k-NN [33] is a non-parametric, instance-based learning algorithm that classifies a new data point based on the majority class among its “k” nearest neighbors. It does not assume any underlying distribution of the data, making it versatile. However, k-NN can be computationally expensive with large datasets, and it is sensitive to the choice of “k” and the distance metric. It also tends to struggle with noisy data, as each neighbor’s class significantly impacts the result.

Decision tree [34] is a tree-structured model that splits the data into subsets based on feature values, making decisions using a series of if-then conditions. Each internal node represents a feature, each branch represents a decision rule, and each leaf node represents an outcome (class label). Decision trees are highly interpretable but prone to overfitting, especially if the tree grows too deep. They are non-linear models and work well for both classification and regression tasks.

Random forest [35] is an ensemble learning method that builds multiple decision trees during training and merges their predictions to improve accuracy and reduce overfitting. Each tree is built using a random subset of features and data points, which introduces variability and makes the model more robust than a single decision tree. Random forest can handle large datasets with higher dimensionality but may become less interpretable due to the combination of multiple trees.

2.2. DL

ConvNeXtBase [22] is a modern convolutional neural network (CNN) architecture designed as an evolution of traditional CNNs, integrating ideas from transformer models while maintaining the efficiency and simplicity of CNNs. Developed by Facebook AI Research, it adopts design principles from the vision transformer (ViT), such as using large kernel sizes and layer normalization, while refining conventional CNN elements. ConvNeXtBase is known for its strong performance in image classification tasks, achieving competitive results with transformers on large-scale datasets like ImageNet, while being more computationally efficient for vision tasks. This model balances high accuracy and efficiency, making it suitable for a variety of visual recognition applications.

ConvNeXtSmall [22] is a variant of the ConvNeXt architecture, designed to offer a more lightweight and efficient model for image classification and other vision tasks. Like ConvNeXtBase, it builds on traditional CNN structures while incorporating innovations inspired by transformer models, such as large kernel sizes and layer normalization. Con-

vNeXtSmall maintains a balance between computational efficiency and accuracy, making it well-suited for scenarios where lower computational cost is important, but strong performance on tasks like object recognition is still required. It delivers competitive results on benchmarks like ImageNet, offering a good trade-off between speed and performance for real-world applications.

ConvNeXtLarge [22] is a larger and more powerful variant of the ConvNeXt architecture, designed for high-performance image classification and visual recognition tasks. Like its smaller counterparts, ConvNeXtLarge combines the strengths of traditional convolutional neural networks with design elements from vision transformers, such as large kernel sizes, layer normalization, and attention mechanisms. It is more computationally intensive but delivers top-tier performance, particularly on large-scale datasets like ImageNet. ConvNeXtLarge is particularly well-suited for tasks that demand higher accuracy and robustness, often outperforming other models in terms of precision while maintaining the efficiency associated with CNN-based architectures.

EfficientNetB0 [23] is the smallest model in the EfficientNet family of convolutional neural networks, designed to achieve high accuracy while being computationally efficient. Developed by Google, EfficientNetB0 uses a novel compound scaling method that uniformly scales network depth, width, and resolution to optimize performance. This approach allows the model to achieve excellent results on benchmarks like ImageNet with significantly fewer parameters and lower computational costs compared to traditional architectures. EfficientNetB0 is particularly useful in resource-constrained environments where both accuracy and efficiency are essential, making it ideal for mobile and edge applications.

EfficientNetB3 [23] is a larger and more powerful variant within the EfficientNet family, designed to balance accuracy and efficiency using Google's compound scaling method. Like EfficientNetB0, it scales depth, width, and resolution uniformly, but with more layers and higher resolution inputs, leading to improved performance on tasks such as image classification. EfficientNetB3 achieves better accuracy than smaller models like B0 while still being more computationally efficient compared to other architectures of similar accuracy. It is well-suited for scenarios where higher accuracy is needed but computational resources are somewhat limited, offering strong performance on benchmarks like ImageNet with manageable resource requirements.

EfficientNetB7 [23] is the largest and most powerful model in the EfficientNet family, designed for high-performance tasks where maximizing accuracy is a priority. Like other EfficientNet models, it uses Google's compound scaling method to effectively scale up the network's depth, width, and input resolution. EfficientNetB7 offers superior accuracy on image classification benchmarks like ImageNet, often achieving state-of-the-art results. However, due to its larger size and higher computational demands, it requires more resources compared to smaller models like EfficientNetB0 or B3. EfficientNetB7 is best suited for tasks that demand high precision and where computational resources are more readily available.

DenseNet121 [24] is a convolutional neural network model that belongs to the DenseNet (Densely Connected Networks) family, introduced to improve information flow between layers. Unlike traditional architectures where layers are connected sequentially, DenseNet121 connects each layer to every other layer in a feed-forward manner. This dense connectivity encourages feature reuse, reduces the vanishing gradient problem, and improves parameter efficiency, allowing the model to be relatively compact while maintaining high performance. DenseNet121 has 121 layers and is particularly effective for image classification tasks, offering strong results on benchmarks like ImageNet while requiring fewer parameters and less computational power than other deep models, such as ResNet. Its efficient design makes it well-suited for various computer vision applications, especially when computational resources are limited.

InceptionV3 [25] is a deep convolutional neural network architecture that is part of Google's Inception family, known for its efficient use of computational resources. It builds on the earlier versions of Inception by refining the "Inception module" concept, where

multiple convolutional filters of different sizes are applied to the same input in parallel. This allows the network to capture a wider range of spatial features, increasing its ability to recognize complex patterns in images. InceptionV3 introduces additional improvements, such as factorized convolutions (splitting a large convolution into smaller ones), grid reduction techniques, and the use of batch normalization to improve training speed and accuracy. It is highly effective for image classification tasks and has achieved strong results on large-scale datasets like ImageNet, offering a good balance between high accuracy and computational efficiency. InceptionV3 is widely used in real-world applications, such as object detection and image analysis, especially when resources are constrained but high performance is required.

MobileNetV2 [27] is a convolutional neural network architecture designed specifically for efficient performance on mobile and resource-constrained devices. Introduced by Google, it improves on the original MobileNet by using an inverted residual structure and linear bottlenecks, which help to reduce computational cost while maintaining accuracy. The key innovation in MobileNetV2 is the “inverted residual block”, which expands the input channels first, applies depthwise separable convolutions, and then compresses the output back down, allowing for more efficient processing. MobileNetV2 is highly efficient in terms of both speed and memory usage, making it ideal for applications like real-time image recognition on mobile devices or edge computing. Despite its lightweight nature, it performs well on benchmarks like ImageNet, offering a good trade-off between performance and efficiency for mobile AI tasks.

ResNet152 [28] is part of the ResNet family, which introduced the concept of residual connections to address the vanishing gradient problem that commonly occurs in very deep networks. These connections enable the network to “skip” layers, allowing the model to learn residual mappings instead of direct mappings, which improves training efficiency and accuracy as depth increases. ResNet152, with 152 layers, is one of the deeper variants, utilizing bottleneck blocks to reduce the computational cost while preserving model capacity. This design allows ResNet152 to achieve high accuracy on challenging tasks, such as image classification on ImageNet, while remaining computationally efficient compared to earlier deep networks. Its robustness and scalability have made it a popular backbone for various computer vision applications, including object detection and image segmentation.

InceptionResNetV2 [26] is a hybrid convolutional neural network architecture that combines the strengths of the Inception and Residual networks. This model builds on the concepts introduced in both the InceptionV3 and ResNet architectures, aiming to enhance performance on image classification tasks while maintaining computational efficiency.

Xception [29] is a deep convolutional neural network architecture that builds upon the ideas of depthwise separable convolutions, which were popularized by MobileNets. Introduced by François Chollet, Xception stands for “Extreme Inception” and is designed to enhance model efficiency and performance by separating the spatial and channel-wise convolutions. The architecture consists of a series of depthwise separable convolutions followed by pointwise convolutions, allowing the model to learn rich feature representations while significantly reducing the number of parameters compared to traditional convolutional layers. Xception has achieved state-of-the-art results on image classification tasks, particularly on benchmarks like ImageNet. Its design emphasizes efficient computation and parameter utilization, making it well-suited for both high-performance applications and deployment in resource-constrained environments. This architecture has influenced subsequent models and is widely used in various computer vision applications, including image classification, object detection, and transfer learning.

Swin transformers [36] are a novel vision transformer architecture developed by Microsoft Research, designed to enhance performance on a variety of computer vision tasks while maintaining computational efficiency. This architecture utilizes a hierarchical structure that processes images at multiple scales, making it particularly effective for tasks such as image classification, object detection, and semantic segmentation. A key feature of swin transformers is their use of shifted windows, which divides the input image into

non-overlapping windows and performs self-attention within these windows. By shifting the windows at each layer, the model captures cross-window interactions, allowing it to learn relationships between different parts of the image more effectively. Additionally, the hierarchical representation builds feature maps of varying resolutions, capturing both fine and coarse details, while windowed self-attention reduces the computational burden typically associated with global self-attention mechanisms. Swin transformers have achieved state-of-the-art results on benchmark datasets like ImageNet, COCO, and ADE20K, demonstrating their effectiveness across various computer vision applications and representing a significant advancement in applying transformer architectures to vision tasks.

The all-selected DL models each offer unique strengths that make them valuable for a diverse range of tasks. ConvNeXt models (Base, Small, Large) represent a modernized version of traditional CNNs, combining the simplicity of convolutions with updates inspired by ViTs, making them efficient and scalable. EfficientNet (B0, B3, B7) uses a specialized scaling method to balance accuracy and computational efficiency, offering excellent performance with relatively low resource demands. DenseNet121 optimizes feature reuse by connecting each layer to all previous layers, leading to efficient learning with fewer parameters. InceptionV3 and InceptionResNetV2 are designed to capture multi-scale features, with the latter combining Inception's multi-scale processing and ResNet's residual connections for improved gradient flow and deeper model training. MobileNetV2 focuses on lightweight architecture, making it ideal for limited computational resources. ResNet152 is a deep network that uses residual connections to ensure stable training and strong performance, even with considerable depth. Xception builds on Inception by replacing standard convolutions with depthwise separable convolutions, offering both efficiency and high accuracy. Finally, swin transformers introduce a transformer-based architecture that efficiently processes images using hierarchical patching, making it well-suited for capturing long-range dependencies in vision tasks. Together, these models provide a broad spectrum of approaches, covering traditional CNNs, efficient neural network designs, and modern transformer-based architectures. Also, they have been widely adopted and validated on popular benchmark datasets, such as ImageNet.

2.3. Metrics

The models, ConvNeXtBase, ConvNeXtSmall, ConvNeXtLarge, EfficientNetB0, EfficientNetB3, EfficientNetB7, DenseNet121, InceptionV3, InceptionResNetV2, MobileNetV2, ResNet152, and Xception, were fine-tuned using consistent parameter settings. The Adaptive Moment Estimation (ADAM) optimizer [37] was employed with a learning rate of 0.0001, and the models were trained for 20 iterations.

All experiments were conducted on Google Colaboratory, a cloud-based platform that allows users to write and execute Python code for free, widely used by data scientists, machine learning practitioners, and researchers. To ensure balanced training, equal numbers of data samples were used for each macrofungi species during both training and testing. Given the limited number of available images, 4-fold cross-validation was implemented. This widely used technique is particularly beneficial when working with small datasets, as it divides the data into k equal-sized subsets (folds). The model is trained and validated across k iterations, with a different fold used for validation and the remaining $k-1$ folds used for training in each round. At each iteration, the model is trained on most of the data and tested on the reserved fold, providing insight into its performance. This process continues until every fold has been used once for validation. Afterward, the results from all iterations are averaged to give a more reliable estimate of the model's generalization performance. Importantly, the training and testing datasets were kept completely separate, ensuring that no images used for training were included in the testing set.

The effectiveness of all the methods applied was evaluated based on several key performance metrics, including accuracy, which measures the overall correctness of the predictions, precision, which assesses the proportion of correctly identified positive instances out of all predicted positives, and recall, which evaluates the model's ability to

identify actual positive cases. Additionally, the F1-score was used to provide a harmonic mean between recall and precision, giving a balanced measure of the model's performance, especially on imbalanced datasets. Lastly, the AUC (Area Under the ROC Curve) was included to assess the model's capability to distinguish between positive and negative classes across varying decision thresholds, with a higher AUC indicating better discriminatory power. The equations for accuracy, precision, recall and F1-score metrics are presented in Equation (1), Equation (2), Equation (3), and Equation (4), respectively.

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \quad (1)$$

$$\text{Precision} = \frac{\text{TP}}{\text{TN} + \text{FP}} \quad (2)$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \quad (3)$$

$$\text{F1 - score} = \frac{2 \times (\text{Precision} \times \text{Recall})}{\text{Precision} + \text{Recall}} \quad (4)$$

In Equations (1)–(3), the terms TP, FP, TN, and FN are defined as follows [38]:

- TP (true positives) corresponds to instances where a diseased condition is accurately predicted.
- FP (false positives) occurs when a healthy sample is incorrectly identified as diseased.
- TN (true negatives) refers to cases where healthy samples are correctly classified.
- FN (false negatives) pertains to instances where a diseased condition is wrongly classified as healthy.

Additionally, the Grad-CAM (Gradient-weighted Class Activation Mapping) [39] technique was employed to analyze the specific areas of the images that the models prioritized during the inference process. Neural networks are composed of multiple interconnected layers with numerous parameters that are adjusted during training to process and interpret input data. However, understanding how the models generate outputs from the given inputs can be challenging, often leading to a lack of transparency and reduced confidence in their predictions. Grad-CAM is a powerful visualization method designed to interpret the decision-making processes of convolutional neural networks. It is particularly useful in image classification tasks, as it highlights the regions of an image that are most influential in the model's prediction. In Grad-CAM visualizations, red regions represent the areas where the model is most focused, meaning that the network uses these regions to form its prediction. Conversely, blue regions signify areas of lesser importance, indicating that the model pays minimal attention to these parts when making its decision. This technique enhances the interpretability of CNNs, offering insights into how the model processes visual data and providing a clearer understanding of its decision-making process.

3. Results

This section outlines the experiments carried out and the results obtained from the study.

The ML models were carefully configured with parameters to provide a balance between accuracy, robustness, and computational efficiency. The parameter settings for these models were set as follows:

For logistic regression:

- Solver = liblinear
- Maximum iteration = 1000
- Regularization parameter C = 1.0

For SVM:

- Kernel = Radial Basis Function (RBF)
- Regularization parameter C = 1.0

- Probability = true
For k-NN:
 - N neighbors = 5
 - weights = distance
 - metric = Minkowski with $p = 2$
- For decision tree:
 - maximum depth = 10
 - minimum samples split = 10
 - minimum samples leaf = 5
- For random forest:
 - N estimators = 100
 - Maximum depth = 10
 - Minimum samples split = 10
 - Minimum samples leaf = 5
 - Maximum features = sqrt

When Table 2 is examined, logistic regression and SVM emerged as the best-performing models, both achieving an accuracy of 0.60, with balanced precision and recall scores of 0.60. Their F1-scores were slightly lower, at 0.58 and 0.56, respectively, reflecting small trade-offs between precision and recall. Both models showed good discriminatory power, with AUC values of 0.70 for logistic regression and 0.71 for SVM. On the other hand, k-NN performed poorly across all metrics, with an accuracy of only 0.45, a low F1-score of 0.40, and an AUC of 0.50, indicating no better performance than random guessing. The decision tree model also showed weak results, with an accuracy of 0.48 and a similar F1-score of 0.48, though its AUC was slightly higher at 0.53. Random forest performed moderately, with an accuracy of 0.57 and an AUC of 0.60, showing better performance than decision tree but still falling short of logistic regression and SVM. Overall, logistic regression and SVM offered the best trade-off between precision and recall, as well as the strongest ability to distinguish between classes, while k-NN and decision tree struggled with lower accuracy and AUC scores. The overall results indicate that traditional methods struggle to achieve high accuracy or robust class separation in this context. Therefore, they would generally be considered moderate-to-poor performers for this task; more advanced methods, like deep learning, were applied to the problem.

Table 2. Experimental results for ML.

	Accuracy	Precision	Recall	F1-Score	AUC
Logistic regression	0.60	0.60	0.60	0.58	0.70
SVM	0.60	0.60	0.60	0.56	0.71
k-NN	0.45	0.42	0.45	0.40	0.50
Decision tree	0.48	0.47	0.48	0.48	0.53
Random forest	0.57	0.57	0.57	0.54	0.60

In DL algorithms for data preprocessing, all images were resized to a uniform resolution to fit the input requirements of the selected models, ensuring consistency across the dataset. Images were also normalized by scaling pixel values to the [0, 1] range, which helps improve model convergence by ensuring that input features have similar scales. Additionally, the dataset was checked for any corrupted or mislabeled images, and appropriate corrections were made.

Regarding data augmentation, we employed horizontal and vertical flipping, as well as random rotations. These techniques were chosen to introduce variability in the orientation of the fungi images, helping the models learn to recognize fungi from different angles and perspectives. By augmenting the data in this way, we aimed to improve the

model's robustness and generalization capabilities, ensuring it can handle variations in the positioning of fungi in real-world settings. These straightforward augmentation methods were applied during training to prevent overfitting and improve model performance on unseen data.

Following the data augmentation process, a series of experiments were conducted to assess the effectiveness of transfer learning-based models, namely ConvNeXtBase, ConvNeXtSmall, ConvNeXtLarge, EfficientNetB0, EfficientNetB3, EfficientNetB7, DenseNet121, InceptionV3, InceptionResNetV2, MobileNetV2, ResNet152, Xception, and swin transformers, in predicting six different fungi species automatically.

In Figure 2, the validation accuracy graphs of the models are presented. The most successful models from the EfficientNet and ConvNeXt families have been included in the graph.

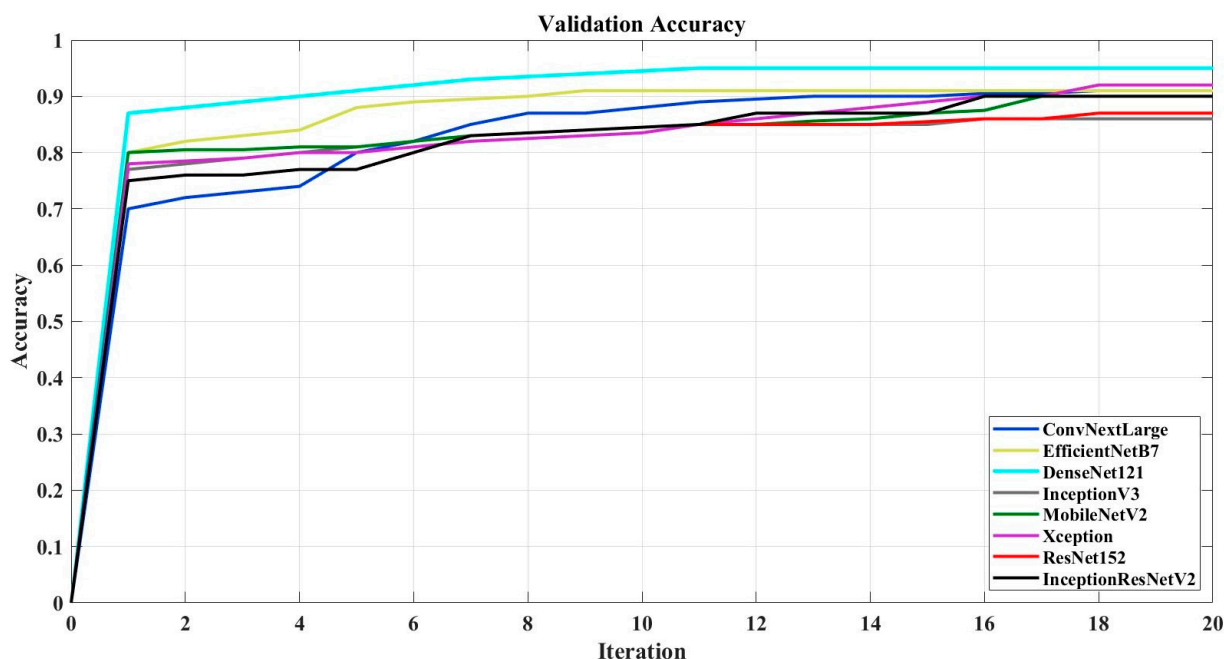


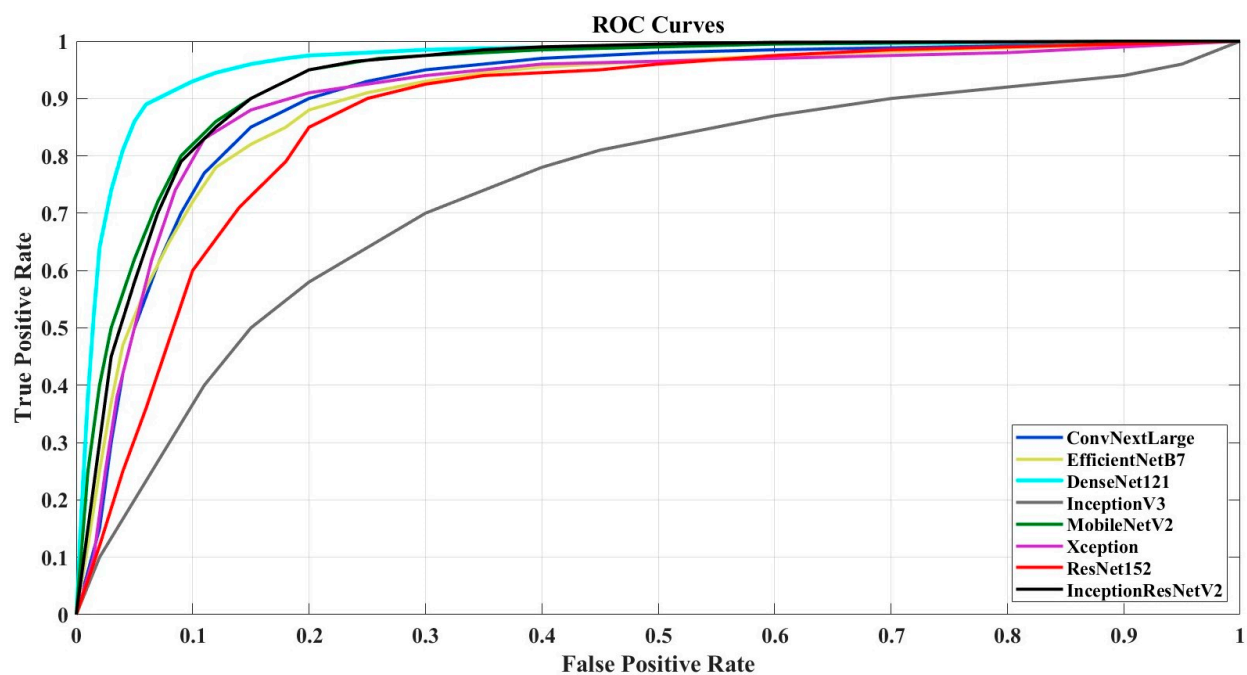
Figure 2. Validation accuracy.

Upon examination of Figure 2, among the evaluated models, DenseNet121 demonstrates the highest validation accuracy, converging rapidly to near-perfect performance, thereby indicating its robustness and efficiency. EfficientNetB7 also achieves commendable performance, attaining a level of accuracy that is highly competitive, though marginally lower than that of DenseNet121. ConvNeXtLarge exhibits strong performance but does not quite attain the validation accuracy observed with DenseNet121 and EfficientNetB7. The remaining models—InceptionV3, MobileNetV2, Xception, ResNet152, and InceptionResNetV2—converge to relatively high accuracy values; however, they fall short when compared to the leading models. Notably, most of the models stabilize in terms of accuracy after approximately 10 to 15 iterations, suggesting that additional training beyond this point yields limited gains. In conclusion, DenseNet121 is distinguished by its superior accuracy, followed closely by EfficientNetB7, while the other architectures achieve comparable but slightly lower levels of performance.

The performance of these models was thoroughly evaluated to determine their accuracy in classification tasks. The results of these experiments are summarized in Table 3, which compares the predictive performance of each model based on various metrics, while Figure 3 displays the ROC (Receiver Operating Characteristic) curves. As in Figure 2, the most successful networks from the ConvNeXt and EfficientNet families are included in this graph.

Table 3. Experimental results for DL.

Models	Accuracy	Precision	Recall	F1-Score	AUC
ConvNeXtBase	0.87	0.87	0.87	0.87	0.87
ConvNeXtSmall	0.88	0.87	0.87	0.87	0.87
ConvNeXtLarge	0.89	0.89	0.90	0.90	0.90
EfficientNetB0	0.82	0.78	0.78	0.79	0.80
EfficientNetB3	0.82	0.78	0.78	0.78	0.80
EfficientNetB7	0.89	0.90	0.88	0.89	0.90
DenseNet121	0.92	0.94	0.90	0.92	0.95
InceptionV3	0.84	0.73	0.73	0.73	0.80
MobileNetV2	0.90	0.93	0.90	0.90	0.92
Xception	0.90	0.92	0.90	0.90	0.90
ResNet152	0.87	0.92	0.82	0.87	0.90
InceptionResNetV2	0.90	0.92	0.87	0.90	0.90
Swin transformers	0.64	0.75	0.60	0.63	0.60

**Figure 3.** ROC curve.

When interpreting Table 3 in the context of classifying six different types of fungi, several insights can be drawn regarding the effectiveness of the various models. DenseNet121 stands out as the top performer, achieving an accuracy of 0.92 and an AUC of 0.95. This indicates that DenseNet121 is particularly effective at distinguishing between different fungi types while minimizing both false positives and false negatives. Its high precision and recall scores further suggest its ability to accurately identify the correct fungi species with a high degree of reliability. The ConvNeXt models also demonstrate strong performance, with ConvNeXtLarge reaching an accuracy of 0.89. The similar performance of ConvNeXtSmall and ConvNeXtBase, both achieving accuracy rates above 0.87, indicates that even the smaller variants of this model are capable of effectively classifying fungi. MobileNetV2 and Xception, each achieving 0.90 accuracy, further underscore their suitability for this classification task. These models strike a balance between high accuracy and computa-

tional efficiency, which is crucial when deploying classification systems in real-world or resource-constrained settings. In contrast, the EfficientNetB0 and B3 models, both scoring 0.82, appear to struggle with accurately classifying fungi images. Their lower precision and recall scores indicate a higher likelihood of misclassifications. However, EfficientNetB7 performs better, with an accuracy of 0.89, suggesting that scaling the EfficientNet architecture can lead to improved performance in fungi classification tasks. InceptionV3 demonstrates a more modest accuracy of 0.84, and its lower precision suggests it may face difficulties in reliably differentiating between various fungi types, making it less ideal for this particular task. ResNet152, with an accuracy of 0.87, proves capable of handling the complexities of fungi classification, making it a reliable option for researchers. However, swin transformers perform poorly, achieving an accuracy of only 0.64. This indicates that this architecture may not be suitable for fungi classification, potentially due to its limitations in capturing the subtle, nuanced features necessary for differentiating fungi types in RGB images.

Considering the ROC curves, DenseNet121 and MobileNetV2 demonstrate the best performance, with curves closest to the top-left corner, indicating a strong balance between high true-positive rates and low false-positive rates. InceptionResNetV2 also performs well, achieving nearly comparable results. EfficientNetB7, ConvNeXtLarge, Xception, and ResNet152 show decent though slightly less effective performance, with their ROC curves positioned further from the top-left corner. InceptionV3 has the weakest ROC curve, indicating a higher false-positive rate and a generally lower ability to distinguish between classes. Overall, DenseNet121 and MobileNetV2 are the top performers, while InceptionV3 is the least effective in this comparison.

High accuracy rates do not necessarily indicate that models are making inferences based on the correct features. To address this, Grad-CAM (Gradient-weighted Class Activation Mapping) visualizations were generated for the three images presented in Figure 4. Figure 5 presents the Grad-CAM heatmaps for ConvNeXt, Figure 6 for EfficientNet, Figure 7 for DenseNet121, InceptionV3, and InceptionResNetV2, and Figure 8 for the MobileNetV2, ResNet152, and Xception models, allowing for an analysis of the specific regions from which the models derive their predictions. These visualizations provide critical insight into the areas of the images that influence the models' classification decisions, facilitating the identification of potential misinterpretations or incorrect focal points.



Figure 4. Images without Grad-CAM visualization.

Upon a detailed examination of the Grad-CAM images, it becomes evidently clear that DenseNet121 emerges as the most effective model for fungi classification. The Grad-CAM visualizations provide insight into the model's ability to focus on the relevant regions of the images, highlighting its superior capability to correctly identify critical features associated with different fungi types. In contrast, the ConvNeXt models, which incorporate transformer-based architectures, exhibit less effective visual attention patterns. This suggests that transformer-based models, as reflected by ConvNeXt's visualizations, are not well-suited for this specific task of fungi classification. The limited efficacy of transformers in this domain is further corroborated by the notably poor performance of the swin transformer model, which struggles to capture the distinguishing features of the fungi images. In comparison, MobileNetV2 demonstrates significant efficacy, ranking just behind DenseNet121. The Grad-CAM visualizations for MobileNetV2 suggest that it successfully focuses on important image regions, supporting its relatively strong performance met-

rics. This indicates that while MobileNetV2 may not reach the same level of accuracy as DenseNet121, it remains a highly competent model for fungi classification, especially in scenarios where computational efficiency is also a priority. Overall, the Grad-CAM analyses provide valuable insight into how these models perform, reinforcing the idea that models like DenseNet121 and MobileNetV2 are more suitable for this classification task than transformer-based architectures such as swin transformer. Swin transformers, like other transformer-based architectures, are designed to capture long-range dependencies by processing images in non-local patches, which works well for tasks with large-scale or global features. However, fungi images often consist of fine-grained textures and intricate local patterns that require precise, detailed feature extraction. CNNs, with their convolutional filters, are inherently better suited for such tasks, as they focus on local features, allowing them to more effectively capture the subtle structural details found in fungi. Additionally, transformer models typically perform optimally with large datasets, where their ability to model long-range dependencies can be fully exploited. Given that the dataset used in this study may not have been large or diverse enough, it is possible that swin transformers struggled to generalize, leading to overfitting or suboptimal performance.

To elucidate the robustness of the DenseNet121 and MobileNetV2 networks, which demonstrated the highest performance metrics, their effectiveness in classifying individual mushroom species and their performance on images contaminated with Gaussian noise were also evaluated.

Gaussian noise was introduced to the images across a Signal-to-Noise Ratio (SNR) range of 10 to 60 (10, 20, 30, 40, 50, 60) [40] to simulate various levels of degradation and evaluate model robustness under differing noise intensities. The detailed assessment of model performance focused primarily on experiments conducted at the highest noise levels within this range, as these levels present a more rigorous and challenging test of the models' capacity to maintain accuracy in adverse conditions. Figure 9 presents images with noise applied at different levels.

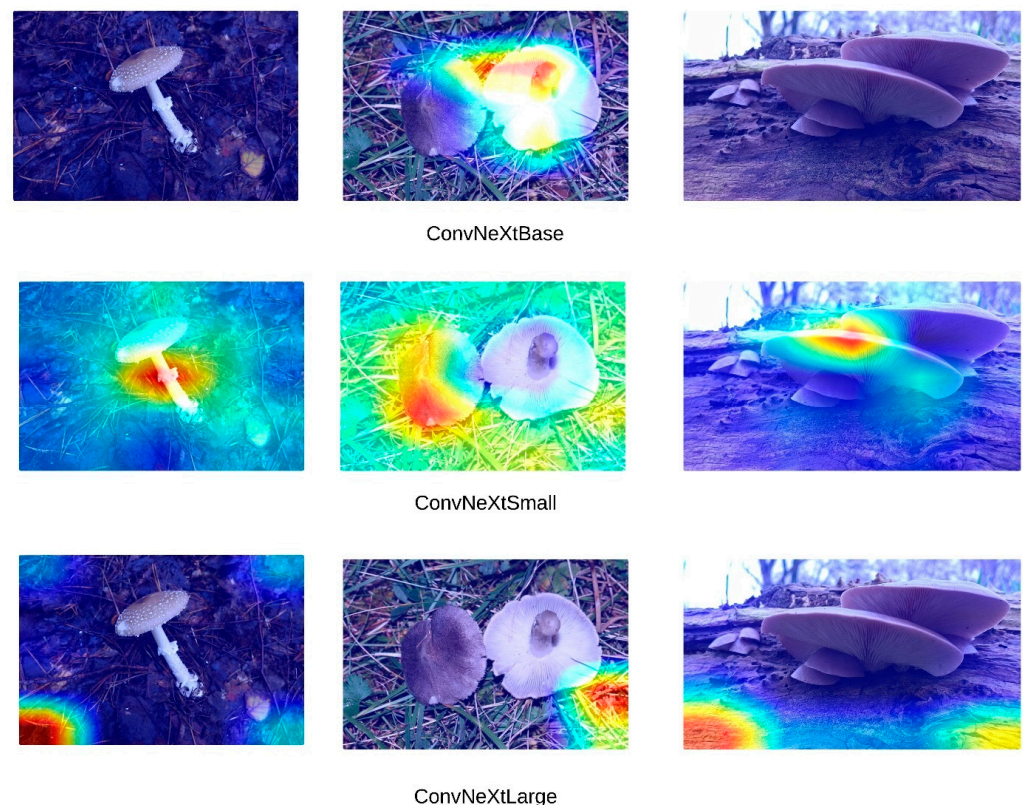


Figure 5. ConvNeXt Grad-CAM visualization.

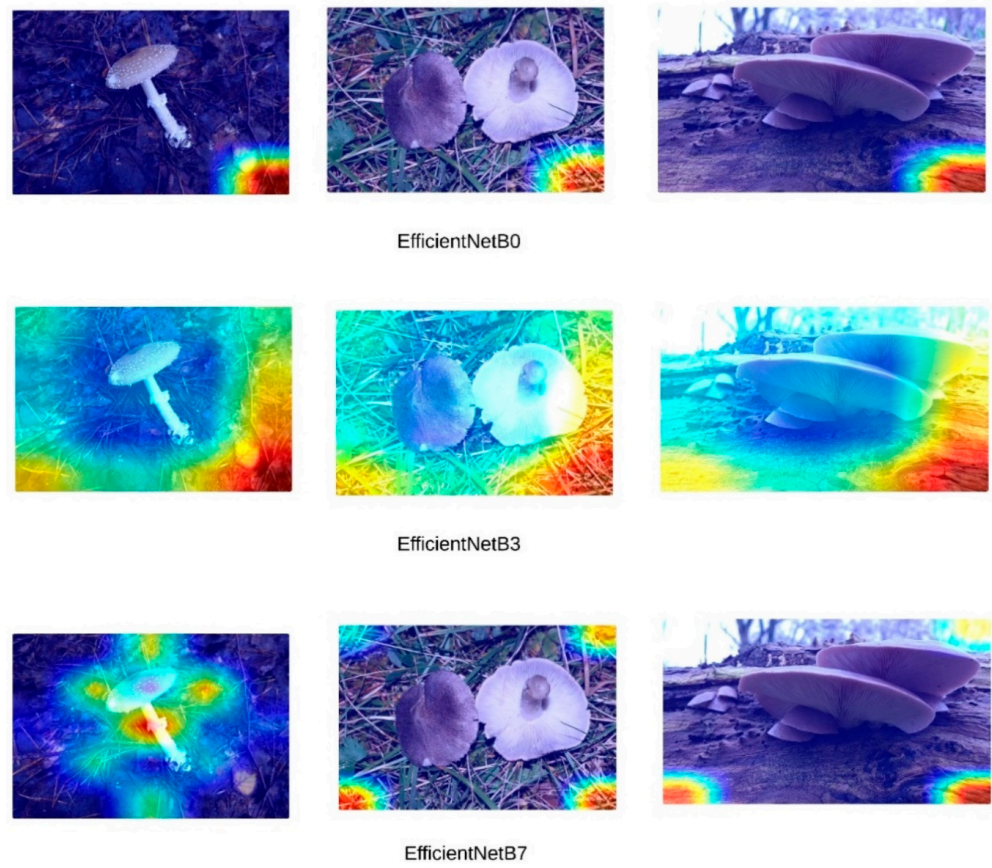


Figure 6. EfficientNet Grad-CAM visualization.

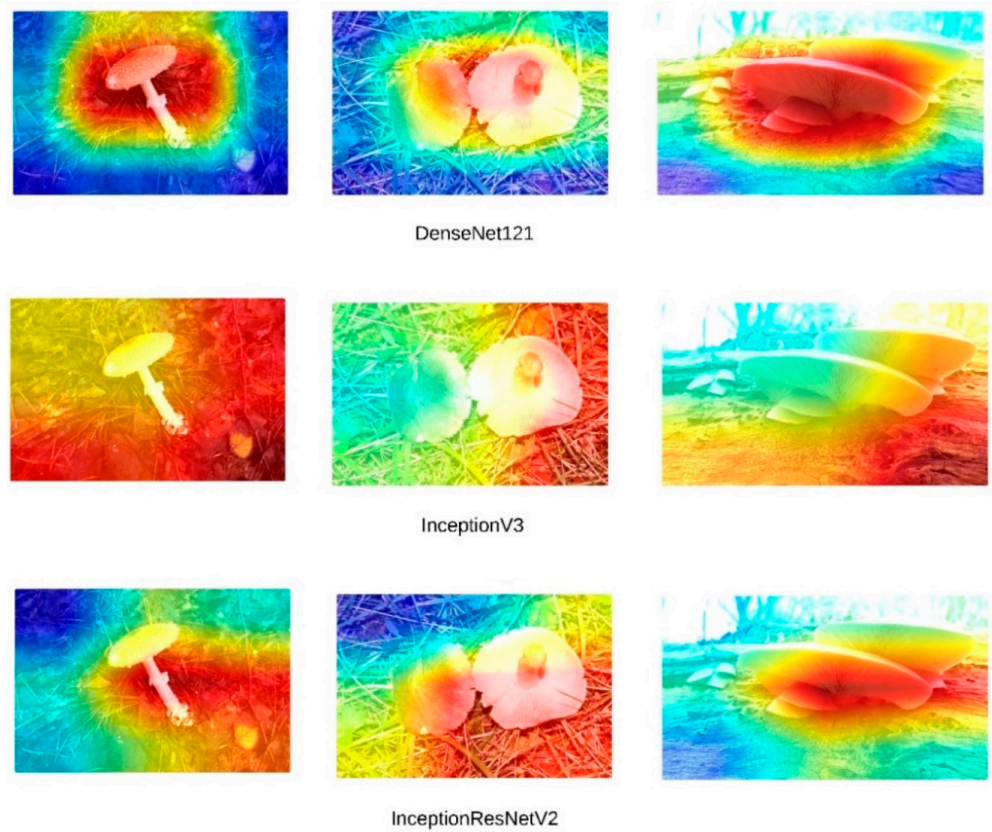


Figure 7. DenseNet121, InceptionV3, and InceptionResNetV2 Grad-CAM visualization.

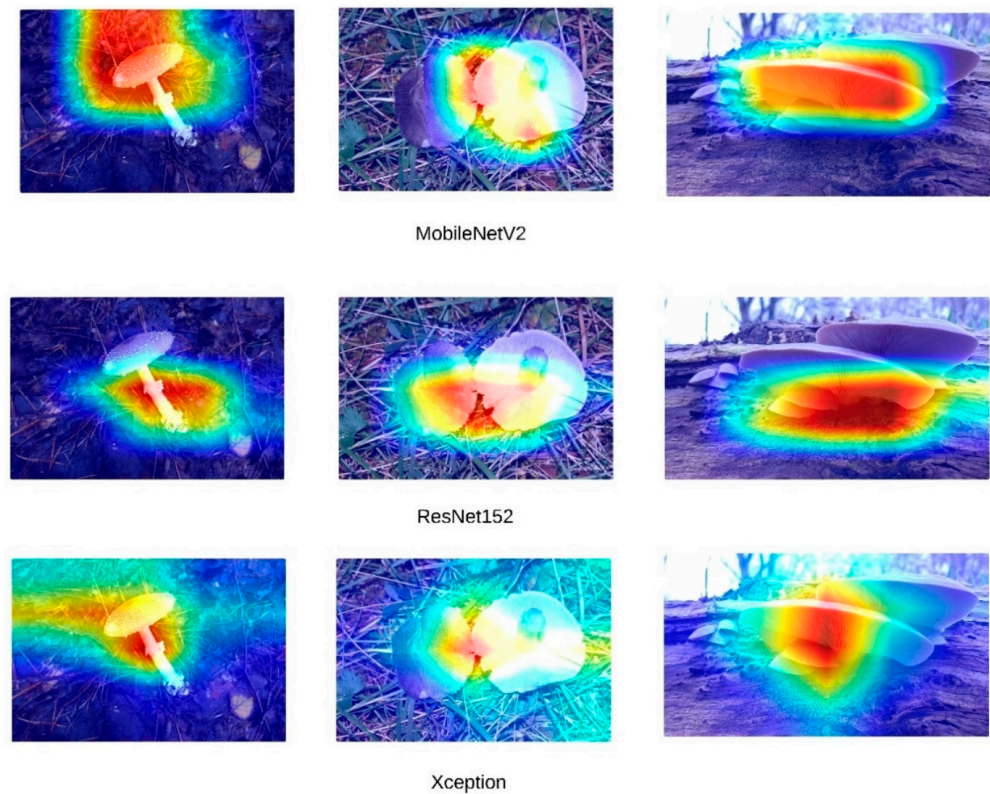


Figure 8. MobileNetV2, ResNet152, and Xception Grad-CAM visualization.

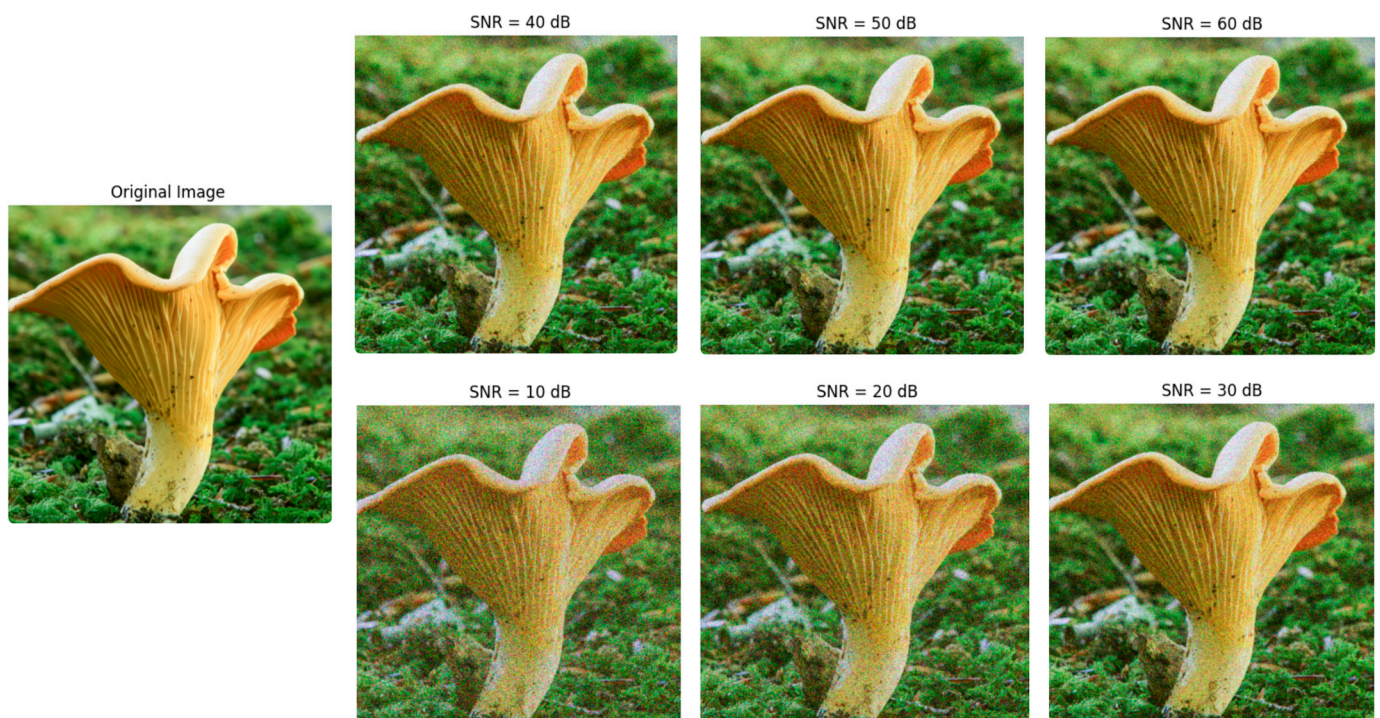


Figure 9. Different levels of Gaussian white noise [40].

The metrics in Table 4 indicate that the models are highly proficient in distinguishing between different fungal species, exhibiting minimal performance variance across categories. This consistency demonstrates the model's suitability for applications requiring accurate and reliable detection and the classification of diverse fungal species.

Table 4. Results of DenseNet121 and MobileNetV2 on fungal species.

	Model	Accuracy	Precision	Recall	F1-Score
<i>Amanita pantherina</i>	DenseNet121	0.92	0.94	0.9	0.92
<i>Boletus edulis</i>		0.918	0.94	0.9	0.92
<i>Cantharellus cibarius</i>		0.92	0.94	0.9	0.918
<i>Lactarius deliciosus</i>		0.92	0.94	0.9	0.92
<i>Pleurotus ostreatus</i>		0.92	0.94	0.9	0.918
<i>Tricholoma terreum</i>		0.92	0.94	0.9	0.92
<i>Amanita pantherina</i>	MobileNetV2	0.9	0.93	0.9	0.9
<i>Boletus edulis</i>		0.899	0.92	0.899	0.899
<i>Cantharellus cibarius</i>		0.899	0.928	0.899	0.899
<i>Lactarius deliciosus</i>		0.9	0.93	0.9	0.9
<i>Pleurotus ostreatus</i>		0.9	0.93	0.9	0.9
<i>Tricholoma terreum</i>		0.9	0.93	0.9	0.9

Since deep networks are inherently capable of performing well on noisy images, noisy images did not need to be included during the training phase. Incorporating noise into the training data could obscure the networks' true robustness by enabling them to learn specific noise patterns, rather than relying on their intrinsic capacity to handle noisy inputs. This approach allows for a more objective evaluation of the models' resilience to noise, as any performance on noisy images in the testing phase reflects the models' innate robustness rather than adaptive learning to noise during training. Evaluation across individual SNR values, ranging from 20 to 60, as well as for all noise levels combined, consistently yielded performance metrics of 0.89 or higher. This consistency across varying noise intensities indicates a high degree of stability in model performance, demonstrating that even in the presence of significant noise interference, the models retained strong accuracy and predictive power.

The performance of DenseNet121 on SNR 10 noisy images, with an accuracy of 0.90, a precision of 0.89, a recall of 0.89, an F1-score of 0.90, and an AUC of 0.91, indicates that the model remains robust and effective even under challenging conditions. The high accuracy and balanced precision and recall demonstrate that DenseNet121 maintains a strong ability to correctly classify noisy inputs while minimizing false positives and false negatives. The F1-score of 0.90 reinforces the model's capacity to handle noisy data effectively, as it shows a good balance between precision and recall. Additionally, the AUC of 0.91 reflects a high degree of separability between classes, indicating that the model can still distinguish between different categories despite the presence of noise. Overall, these results suggest that DenseNet121 is resilient to Gaussian noise and retains a high level of performance, making it suitable for applications where data may not always be clean or noise-free. Similarly, the performance metrics of MobileNetV2 on SNR-10 noisy images indicate a robust classification ability, with an accuracy of 0.87, reflecting that the model correctly identified 87% of instances in the dataset despite the noise. The F1-score, also at 0.87, demonstrates a well-balanced performance, effectively managing both precision and recall, which is crucial for applications where false positives and false negatives are significant. With a precision value of 0.90, the model indicates a high likelihood of correctness when predicting positive classes, thereby minimizing erroneous classifications. Meanwhile, a recall of 0.87 signifies that the model successfully detects 87% of actual positive instances, underscoring its effectiveness in identifying the target class. Finally, an AUC of 0.90 highlights MobileNetV2's strong discriminatory power, illustrating its capability to distinguish between positive and negative instances across different classification thresholds. Collectively, these metrics suggest that MobileNetV2 performs reliably in classification tasks. Figure 10 shows the Grad-CAM colorization of DenseNet121 and MobileNetV2 on SNR-10 noisy images.

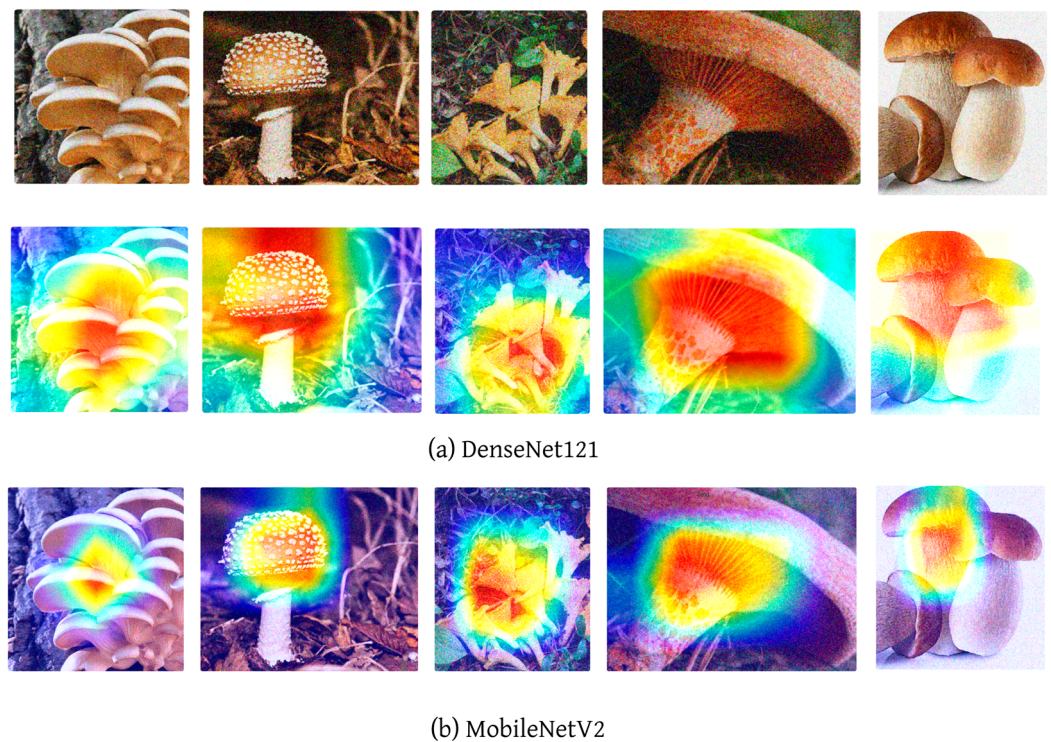


Figure 10. DenseNet121 and MobileNetV2 Grad-CAM visualization on SNR-10 noisy images.

In summary, the results indicate that models DenseNet121 and MobileNetV2 are well-equipped to tackle the challenges of classifying RGB images of fungi, while models such as ML and swin transformers may require further optimization to be viable for this task. Future work could explore whether increasing the dataset size or employing hybrid architectures might help improve the performance of transformer-based models for fungi classification. Furthermore, applying image enhancement [41] techniques to low-quality fungi images could be explored to assess their potential impact on enhancing model performance.

4. Discussion

The results of this study demonstrate the effectiveness of deep learning models in the classification of mushrooms, with findings that align well with the existing literature on similar applications. DenseNet121, EfficientNet, and MobileNet emerged as the key models capable of handling the complexity and heterogeneity of mushroom image data, with DenseNet121 consistently proving the most effective. Specifically, DenseNet121 achieved high reliability across multiple studies, with reported accuracy rates reaching 98% [42–44]. This consistency highlights its strong capacity to capture essential features for accurate fungi classification. The observed performance differences between models, particularly during the training phases, underscore the importance of selecting appropriate architectures for tasks involving complex image data.

Convolutional neural networks (CNNs) have shown significant potential in plant disease detection and similar image classification tasks, and this study affirms their utility for fungal species identification [45–47]. For instance, CNN-based models have previously demonstrated high classification accuracy in detecting diseases such as rice blight, achieving rates up to 97.7% [45]. Such success translates well to the context of fungal infections, suggesting CNN models are powerful tools for agricultural disease management and ecological studies involving fungi [45–47].

A critical factor affecting the performance of deep learning models is the size and diversity of the training dataset. As widely evidenced in previous research, larger and more varied datasets contribute substantially to the accuracy and generalizability of deep

learning models [48–50]. In this study, expanding the dataset through augmentation techniques improved classification accuracy and enhanced the models' robustness. These methods have successfully minimized misclassification rates and facilitated the more efficient detection of fruit body emergence, particularly when the models are trained on constrained datasets. By enhancing accuracy, these approaches ensure that the detection process remains reliable and effective even with limited data [51,52].

In addition to deep learning, methods such as transfer learning and ensemble modeling provide significant advantages in fungal classification and detection. By leveraging pre-trained models on large datasets, transfer learning enables accurate classification even when training data are scarce. It has been noted that models using transfer learning often outperform those trained from scratch by a margin of 10–15%, consistent with the results obtained here [53–55]. Ensemble modeling approaches also showed significant promise; combining multiple models (e.g., the DEX ensemble comprising DenseNet121, EfficientNetB7, and Xception) reduced classification errors and achieved accuracy up to 98%, enhancing the precision of ecological monitoring [53–59].

DenseNet121 emerged as the most effective model in this study, achieving the highest accuracy of 0.92 and an AUC of 0.95, indicating its superior ability to distinguish between different types of fungi with minimal false positives and negatives [43–58]. Its strong precision and recall metrics further reinforce its utility in accurately identifying fungi species, making it highly reliable for classification tasks. Similarly, ConvNeXtLarge also demonstrated solid performance, although its overall accuracy of 0.89 fell slightly behind DenseNet121 [59–62]. Visual analyses using Grad-CAM, however, indicated that ConvNeXt models exhibited less effective attention on crucial features, particularly compared to DenseNet121. This highlights a limitation of current transformer-based architectures, as they may lack the sensitivity required for detailed feature extraction in fungal image classification [63,64].

The performance of transformer-based models varied significantly. While ConvNeXtSmall and ConvNeXtBase showed moderate success, the swin transformer performed poorly, achieving an accuracy of only 0.64 [65–67]. These findings suggest that, in their current form, transformer architectures are not as suitable for this task compared to CNN-based models. Further optimizations or architectural modifications might be necessary to enhance their applicability in fungal classification [67–75].

Among the CNN models tested, MobileNetV2 demonstrated high performance with an accuracy of 0.90. Its computational efficiency makes it an attractive option for practical use, particularly in resource-constrained environments [75–78]. The Grad-CAM visualizations for MobileNetV2 indicated its capability to focus on relevant regions in the images, reinforcing its efficacy for the task. Xception also performed well, with an accuracy of 0.90, though other models like InceptionV3, with an accuracy of only 0.84, struggled to match this level of performance [79–81]. The EfficientNet family showed mixed results; while EfficientNetB7 achieved an accuracy of 0.89, models like EfficientNetB0 and EfficientNetB3 performed less effectively, with an accuracy of around 0.82 [82–85].

ResNet152, achieving an accuracy of 0.87, demonstrated its ability to handle the complexities of the dataset, although it was outperformed by more recent models such as DenseNet121 and MobileNetV2 [86]. This suggests that newer architectures with more advanced feature extraction capabilities provide better solutions for complex biological imagery tasks [87].

In summary, this study highlights the effectiveness of DenseNet121 and MobileNetV2 for fungi classification, as evidenced by their quantitative performance and visual attention analyses using Grad-CAM [88]. DenseNet121's superior ability to capture crucial image features, coupled with MobileNetV2's computational efficiency, makes them highly suitable for practical deployment. However, the limited effectiveness of traditional ML techniques and transformer-based architectures, particularly the swin transformer, indicates the need for further refinement or optimization in this domain.

5. Conclusions

This study investigated the application of deep learning models for classifying six different macrofungi species using RGB images. The comparative analysis demonstrated that DenseNet121 was the most effective model, achieving the highest accuracy (92%) and AUC (95%) scores. MobileNetV2 also proved to be a strong contender, offering a good balance of performance and computational efficiency. These results indicate that DenseNet121 and MobileNetV2 are well-suited for practical fungi classification tasks, particularly in environments with limited computational resources.

Transformer-based models, however, particularly the swin transformer, did not perform as well as CNN-based models in this study. Their relatively poor performance underscores transformers' challenges in distinguishing between complex fungal features in RGB images. This suggests they may require further optimization or modification for practical use in this domain.

For future research, expanding and diversifying the dataset is recommended to improve the generalizability of the models. The use of ensemble learning techniques, combining the strengths of different models, could further enhance classification accuracy. Moreover, integrating additional data types, such as biochemical properties, scanning electron microscopy (SEM), and light microscopy images, may offer deeper insights into fungal features, improving model performance. Incorporating RNA/DNA sequencing data would also allow for genetic-level identification, providing a more comprehensive understanding beyond just morphological classification. These advancements could lead to more precise and reliable methods for ecological monitoring and agricultural management of fungi.

Author Contributions: Conceptualization, S.O., F.E. and E.K.; Methodology, M.S.G. and I.A.; Software, E.O., K.A. and T.A.; Validation, M.S.G., I.A. and F.E.; Formal Analysis, S.O. and E.K.; Investigation, F.E. and K.A.; Resources, M.S.G., F.E., K.A., I.A. and T.A.; Writing—Original Draft Preparation, S.O., I.A. and F.E.; Writing—Review and Editing, M.S.G. and E.O.; Visualization, S.O. and F.E.; Supervision, M.S.G., I.A. and F.E. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The raw data supporting the conclusions of this article will be made available by the authors on request.

Acknowledgments: We would like to thank "Teus Teknologi A.S." for providing the technical infrastructure that enabled us to obtain data for the artificial intelligence algorithms used in this study.

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

References

1. Lu, H.; Lou, H.; Hu, J.; Liu, Z.; Chen, Q. Macrofungi: A Review of Cultivation Strategies, Bioactivity, and Application of Mushrooms. *Compr. Rev. Food Sci. Food Saf.* **2020**, *19*, 2333–2356. [[CrossRef](#)] [[PubMed](#)]
2. Niego, A.G.T.; Rapior, S.; Thongklang, N.; Raspé, O.; Hyde, K.D.; Mortimer, P. Reviewing the Contributions of Macrofungi to Forest Ecosystem Processes and Services. *Fungal Biol. Rev.* **2023**, *44*, 100294. [[CrossRef](#)]
3. Ache, N.T.; Manju, E.B.; Lawrence, M.N.; Tonjock, R.K. Nutrient and Mineral Components of Wild Edible Mushrooms from the Kilum-Ijim Forest, Cameroon. *Afr. J. Food Sci.* **2021**, *15*, 152–161. [[CrossRef](#)]
4. He, M.-Q.; Wang, M.-Q.; Chen, Z.-H.; Deng, W.-Q.; Li, T.-H.; Vizzini, A.; Jeewon, R.; Hyde, K.D.; Zhao, R.-L. Potential Benefits and Harms: A Review of Poisonous Mushrooms in the World. *Fungal Biol. Rev.* **2022**, *42*, 56–68. [[CrossRef](#)]
5. Mortimer, P.E.; Boa, E.; Hyde, K.D. Macrofungi as Food. In *Encyclopedia of Mycology*; Elsevier: Amsterdam, The Netherlands, 2021; pp. 405–417.
6. Satora, L.; Pach, D.; Ciszowski, K.; Winnik, L. Panther Cap *Amanita pantherina* Poisoning Case Report and Review. *Toxicol.* **2006**, *47*, 605–607. [[CrossRef](#)] [[PubMed](#)]

7. Tsujikawa, K.; Kuwayama, K.; Miyaguchi, H.; Kanamori, T.; Iwata, Y.; Inoue, H.; Kishi, T. Determination of Muscimol and Ibotenic Acid in *Amanita* Mushrooms by High-Performance Liquid Chromatography and Liquid Chromatography-Tandem Mass Spectrometry. *J. Chromatogr. B* **2007**, *852*, 430–435. [[CrossRef](#)]
8. Tan, Y.; Zeng, N.K.; Xu, B. Chemical Profiles and Health-Promoting Effects of Porcini Mushroom (*Boletus edulis*): A Narrative Review. *Food Chem.* **2022**, *390*, 133199. [[CrossRef](#)] [[PubMed](#)]
9. Moghaddam, M.; Ghobad-Nejhad, M.; Stegemann, T.; Çiçek, S.S.; Zidorn, C.; Javanmard, M. Nutritional Composition and Odor-Contributing Volatile Compounds of the Edible Mushroom *Cantharellus alborufescens*. *Molecules* **2023**, *28*, 7516. [[CrossRef](#)]
10. Su, Z.; Xu, B. Chemical Compositions and Health-Promoting Effects of Wild Edible Mushroom Milk-Cap (*Lactarius deliciosus*): A Review. *Food Biosci.* **2024**, *62*, 105118. [[CrossRef](#)]
11. Nakazawa, T.; Kawauchi, M.; Otsuka, Y.; Han, J.; Koshi, D.; Schiphof, K.; Honda, Y. *Pleurotus ostreatus* as a Model Mushroom in Genetics, Cell Biology, and Material Sciences. *Appl. Microbiol. Biotechnol.* **2024**, *108*, 217. [[CrossRef](#)]
12. Törös, G.; El-Ramady, H.; Béni, Á.; Peles, F.; Gulyás, G.; Czeglédi, L.; Prokisch, J. *Pleurotus ostreatus* Mushroom: A Promising Feed Supplement in Poultry Farming. *Agriculture* **2024**, *14*, 663. [[CrossRef](#)]
13. Davoli, P.; Floriani, M.; Assisi, F.; Kob, K.; Sitta, N. Comment on “Chemical and Toxicological Investigations of a Previously Unknown Poisonous European Mushroom *Tricholoma terreum*”. *Chem. Eur. J.* **2016**, *22*, 5786–5788. [[CrossRef](#)]
14. Picek, L.; Šulc, M.; Matas, J.; Heilmann-Clausen, J.; Jeppesen, T.S.; Lind, E. Automatic Fungi Recognition: Deep Learning Meets Mycology. *Sensors* **2022**, *22*, 633. [[CrossRef](#)] [[PubMed](#)]
15. Yan, Z.; Liu, H.; Li, J.; Wang, Y. Application of Identification and Evaluation Techniques for Edible Mushrooms: A Review. *Crit. Rev. Anal. Chem.* **2023**, *53*, 634–654. [[CrossRef](#)] [[PubMed](#)]
16. Chaturika, K.; Siriwardena, E.; Bandara, H.; Perera, G.; Dilshanka, K. Developing an Identification System for Different Types of Edible Mushrooms in Sri Lanka Using Machine Learning and Image Processing. *Int. J. Eng. Manag. Res.* **2023**, *13*, 54–59.
17. Bartlett, P.; Eberhardt, U.; Schütz, N.; Beker, H.J. Species Determination Using AI Machine-Learning Algorithms: *Hebeloma* as a Case Study. *IMA Fungus* **2022**, *13*, 13. [[CrossRef](#)]
18. McCulloch, W.S.; Pitts, W. A Logical Calculus of the Ideas Immanent in Nervous Activity. *Bull. Math. Biophys.* **1943**, *5*, 115–133. [[CrossRef](#)]
19. Rosenblatt, F. The Perceptron: A Probabilistic Model for Information Storage and Organization in the Brain. *Psychol. Rev.* **1958**, *65*, 386. [[CrossRef](#)] [[PubMed](#)]
20. Hinton, G.E. Learning Multiple Layers of Representation. *Trends Cogn. Sci.* **2007**, *11*, 428–434. [[CrossRef](#)]
21. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. Imagenet Classification with Deep Convolutional Neural Networks. In Proceedings of the 26th Annual Conference on Neural Information Processing Systems 2012, Lake Tahoe, NV, USA, 3–6 December 2012.
22. Liu, Z.; Mao, H.; Wu, C.-Y.; Feichtenhofer, C.; Darrell, T.; Xie, S. A Convnet for the 2020s. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022.
23. Tan, M. Efficientnet: Rethinking Model Scaling for Convolutional Neural Networks. *arXiv* **2019**, arXiv:1905.11946.
24. Huang, G.; Liu, Z.; Van Der Maaten, L.; Weinberger, K.Q. Densely Connected Convolutional Networks. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017.
25. Szegedy, C.; Vanhoucke, V.; Ioffe, S.; Shlens, J.; Wojna, Z. Rethinking the Inception Architecture for Computer Vision. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016.
26. Szegedy, C.; Ioffe, S.; Vanhoucke, V.; Alemi, A. Inception-v4, Inception-ResNet and the Impact of Residual Connections on Learning. In Proceedings of the 31st AAAI Conference on Artificial Intelligence, San Francisco, CA, USA, 4–9 February 2017.
27. Sandler, M.; Howard, A.; Zhu, M.; Zhmoginov, A.; Chen, L.-C. MobileNetV2: Inverted Residuals and Linear Bottlenecks. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018.
28. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep Residual Learning for Image Recognition. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016.
29. Chollet, F. Xception: Deep Learning with Depthwise Separable Convolutions. In Proceedings of the 30th IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017.
30. Liu, Z.; Lin, Y.; Cao, Y.; Hu, H.; Wei, Y.; Zhang, Z.; Guo, B. Swin Transformer: Hierarchical Vision Transformer Using Shifted Windows. In Proceedings of the 2021 IEEE/CVF International Conference on Computer Vision Workshops (ICCVW), Montreal, BC, Canada, 11–17 October 2021.
31. Cox, D.R. The Regression Analysis of Binary Sequences. *J. R. Stat. Soc. Ser. B* **1958**, *20*, 215–232. [[CrossRef](#)]
32. Cortes, C. Support-Vector Networks. *Mach. Learn.* **1995**, *20*, 273–297. [[CrossRef](#)]
33. Fix, E. *Discriminatory Analysis: Nonparametric Discrimination, Consistency Properties*; USAF School of Aviation Medicine: Dayton, OH, USA, 1985; Volume 1.
34. Quinlan, J. Induction of Decision Trees. *Mach. Learn.* **1986**, *1*, 81–106. [[CrossRef](#)]
35. Breiman, L. Random Forests. *Mach. Learn.* **2001**, *45*, 5–32. [[CrossRef](#)]
36. Liu, Z.; Lin, Y.; Cao, Y.; Hu, H.; Wei, Y.; Zhang, Z.; Lin, S.; Guo, B. Swin Transformer: Hierarchical Vision Transformer Using Shifted Windows. In Proceedings of the International Conference on Computer Vision (ICCV), Montreal, QC, Canada, 10–17 October 2021; pp. 10012–10022.
37. Kingma, D.P. Adam: A Method for Stochastic Optimization. *arXiv* **2014**, arXiv:1412.6980.

38. Ozsari, S.; Yapicioglu, F.R.; Yilmaz, D.; Kamburoglu, K.; Guzel, M.S.; Bostanci, G.E.; Asuroglu, T. Interpretation of Magnetic Resonance Images of Temporomandibular Joint Disorders by Using Deep Learning. *IEEE Access* **2023**, *11*, 49102–49113. [[CrossRef](#)]
39. Selvaraju, R.R.; Cogswell, M.; Das, A.; Vedantam, R.; Parikh, D.; Batra, D. Grad-CAM: Visual Explanations from Deep Networks via Gradient-Based Localization. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017.
40. Cheng, X.; Huo, Y.; Lin, S.; Dong, Y.; Zhao, S.; Zhang, M.; Wang, H. Deep Feature Aggregation Network for Hyperspectral Anomaly Detection. *IEEE Trans. Instrum. Meas.* **2024**, *73*, 5033016. [[CrossRef](#)]
41. Li, J.; Zheng, K.; Liu, W.; Li, Z.; Yu, H.; Ni, L. Model-Guided Coarse-to-Fine Fusion Network for Unsupervised Hyperspectral Image Super-Resolution. *IEEE Geosci. Remote Sens. Lett.* **2023**, *20*, 5508605. [[CrossRef](#)]
42. Crespo-Michel, A.; Alonso-Arévalo, M.A.; Hernández-Martínez, R. Developing a Microscope Image Dataset for Fungal Spore Classification in Grapevine Using Deep Learning. *J. Agric. Food Res.* **2023**, *14*, 100805. [[CrossRef](#)]
43. Görlich, F.; Marks, E.; Mahlein, A.K.; König, K.; Lottes, P.; Stachniss, C. UAV-Based Classification of Cercospora Leaf Spot Using RGB Images. *Drones* **2021**, *5*, 34. [[CrossRef](#)]
44. Ahad, M.T.; Li, Y.; Song, B.; Bhuiyan, T. Comparison of CNN-Based Deep Learning Architectures for Rice Diseases Classification. *Artif. Intell. Agric.* **2023**, *9*, 22–35. [[CrossRef](#)]
45. Struniawski, K.; Kozera, R.; Trzcinski, P.; Lisek, A.; Paszt, L.S. Automated Identification of Soil Fungi and Chromista through Convolutional Neural Networks. *Eng. Appl. Artif. Intell.* **2024**, *127*, 107333. [[CrossRef](#)]
46. Abade, A.; Porto, L.F.; Ferreira, P.A.; de Barros Vidal, F. NemaNet: A Convolutional Neural Network Model for Identification of Soybean Nematodes. *Biosyst. Eng.* **2022**, *213*, 39–62. [[CrossRef](#)]
47. Wang, X.; Polder, G.; Focker, M.; Liu, C. Sága, a Deep Learning Spectral Analysis Tool for Fungal Detection in Grains—A Case Study to Detect Fusarium in Winter Wheat. *Toxins* **2024**, *16*, 354. [[CrossRef](#)]
48. Jaiswal, G.; Sharma, A.; Yadav, S.K. Critical Insights into Modern Hyperspectral Image Applications Through Deep Learning. *Wiley Interdiscip. Rev. Data Min. Knowl. Discov.* **2021**, *11*, e1426. [[CrossRef](#)]
49. Khan, M.A.; Ali, M.; Shah, M.; Mahmood, T.; Ahmad, M.; Jhanjhi, N.Z.; Jaha, E.S. Machine Learning-Based Detection and Classification of Walnut Fungi Diseases. *Intell. Autom. Soft Comput.* **2021**, *30*, 771–785. [[CrossRef](#)]
50. Hassan, E.; Talaat, F.M.; Adel, S.; Abdelrazek, S.; Aziz, A.; Nam, Y.; El-Rashidy, N. Robust Deep Learning Model for Black Fungus Detection Based on Gabor Filter and Transfer Learning. *Comput. Syst. Sci. Eng.* **2023**, *47*, 1508–1525. [[CrossRef](#)]
51. Rahman, M.A.; Clinch, M.; Reynolds, J.; Dangott, B.; Villegas, D.M.M.; Nassar, A.; Akkus, Z. Classification of Fungal Genera from Microscopic Images Using Artificial Intelligence. *J. Pathol. Inform.* **2023**, *14*, 100314. [[CrossRef](#)]
52. Genaev, M.A.; Skolotneva, E.S.; Gulyaeva, E.I.; Orlova, E.A.; Bechtold, N.P.; Afonnikov, D.A. Image-Based Wheat Fungi Diseases Identification by Deep Learning. *Plants* **2021**, *10*, 1500. [[CrossRef](#)]
53. Giakoumoglou, N.; Pechlivani, E.M.; Sakelliou, A.; Klaridopoulos, C.; Frangakis, N.; Tzovaras, D. Deep Learning-Based Multi-Spectral Identification of Grey Mould. *Smart Agric. Technol.* **2023**, *4*, 100174. [[CrossRef](#)]
54. Subasi, A.; Mitra, A.; Ozyurt, F.; Tuncer, T. Automated COVID-19 Detection from CT Images Using Deep Learning. In *Computer-Aided Design and Diagnosis Methods for Biomedical Applications*; CRC Press: Boca Raton, FL, USA, 2021; pp. 153–176.
55. Visuña, L.; Yang, D.; Garcia-Blas, J.; Carretero, J. Computer-Aided Diagnostic for Classifying Chest X-Ray Images Using Deep Ensemble Learning. *BMC Med. Imaging* **2022**, *22*, 178. [[CrossRef](#)] [[PubMed](#)]
56. Kumar, S.J.; Kanna, G.P.; Raja, D.P.; Kumar, Y. A Comprehensive Study on Deep Learning Models for the Detection of Ovarian Cancer and Glomerular Kidney Disease Using Histopathological Images. *Arch. Comput. Methods Eng.* **2024**, 1–27. [[CrossRef](#)]
57. Deng, R.; Tao, M.; Xing, H.; Yang, X.; Liu, C.; Liao, K.; Qi, L. Automatic Diagnosis of Rice Diseases Using Deep Learning. *Front. Plant Sci.* **2021**, *12*, 701038. [[CrossRef](#)] [[PubMed](#)]
58. Bakr, M.; Abdel-Gaber, S.; Nasr, M.; Hazman, M. DenseNet Based Model for Plant Diseases Diagnosis. *Eur. J. Electr. Eng. Comput. Sci.* **2022**, *6*, 1–9. [[CrossRef](#)]
59. Raza, A.; Rustam, F.; Siddiqui, H.U.R.; Diez, I.D.L.T.; Ashraf, I. Predicting Microbe Organisms Using Data of Living Micro Forms of Life and Hybrid Microbes Classifier. *PLoS ONE* **2023**, *18*, e0284522. [[CrossRef](#)]
60. Suciú, C.I.; Marginean, A.; Suciú, V.I.; Muntean, G.A.; Nicoară, S.D. Diabetic Macular Edema Optical Coherence Tomography Biomarkers Detected with EfficientNetV2B1 and ConvNeXt. *Diagnostics* **2023**, *14*, 76. [[CrossRef](#)] [[PubMed](#)]
61. Huynh, H.N.; Tran, A.T.; Tran, T.N. Region-of-interest optimization for deep-learning-based breast cancer detection in mammograms. *Appl. Sci.* **2023**, *13*, 6894. [[CrossRef](#)]
62. Lau, K.W.; Po, L.M.; Rehman, Y.A.U. Large separable kernel attention: Rethinking the large kernel attention design in CNN. *Expert Syst. Appl.* **2024**, *236*, 121352. [[CrossRef](#)]
63. Behrendt, F.; Sonawane, S.; Bhattacharya, D.; Maack, L.; Krüger, J.; Opfer, R.; Schlaefer, A. Quantitative evaluation of activation maps for weakly-supervised lung nodule segmentation. In Proceedings of the Medical Imaging 2024: Computer-Aided Diagnosis, San Diego, CA, USA, 18–22 February 2024; Volume 12927, pp. 643–646.
64. Yu, X.; Wang, S.; Hu, J. Guided Random Mask: Adaptively Regularizing Deep Neural Networks for Medical Image Analysis by Potential Lesions. *Appl. Sci.* **2022**, *12*, 9099. [[CrossRef](#)]
65. Zhu, W.; Zhang, H.; Zhang, C.; Zhu, X.; Guan, Z.; Jia, J. Surface defect detection and classification of steel using an efficient Swin Transformer. *Adv. Eng. Inform.* **2023**, *57*, 102061. [[CrossRef](#)]

66. Jamali, A.; Mahdianpari, M. Swin transformer and deep convolutional neural networks for coastal wetland classification using sentinel-1, sentinel-2, and LiDAR data. *Remote Sens.* **2022**, *14*, 359. [[CrossRef](#)]
67. He, X.; Zhou, Y.; Zhao, J.; Zhang, D.; Yao, R.; Xue, Y. Swin transformer embedding UNet for remote sensing image semantic segmentation. *IEEE Trans. Geosci. Remote Sens.* **2022**, *60*, 4408715. [[CrossRef](#)]
68. Dang, M.; Wang, H.; Li, Y.; Nguyen, T.H.; Tightiz, L.; Xuan-Mung, N.; Nguyen, T.N. Computer Vision for Plant Disease Recognition: A Comprehensive Review. *Bot. Rev.* **2024**, *90*, 251–311. [[CrossRef](#)]
69. Watson, E.; Viana, T.; Zhang, S. Augmented behavioral annotation tools, with application to multimodal datasets and models: A systematic review. *AI* **2023**, *4*, 128–171. [[CrossRef](#)]
70. Alzahrani, S.M.; Qahtani, A.M. Knowledge distillation in transformers with tripartite attention: Multiclass brain tumor detection in highly augmented MRIs. *J. King Saud Univ. Comput. Inf. Sci.* **2024**, *36*, 101907. [[CrossRef](#)]
71. Yang, J.; Liu, J.; Xu, N.; Huang, J. TVT: Transferable vision transformer for unsupervised domain adaptation. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, Waikoloa, HI, USA, 2–7 January 2023; pp. 520–530.
72. Guefrachi, S.; Ecthioui, A.; Hamam, H. Diabetic Retinopathy Detection Using Deep Learning Multistage Training Method. *Arabian J. Sci. Eng.* **2024**, 1–18. [[CrossRef](#)]
73. Luján-García, J.E.; Villuendas-Rey, Y.; López-Yáñez, I.; Camacho-Nieto, O.; Yáñez-Márquez, C. Nanochest-net: A simple convolutional network for radiological studies classification. *Diagnostics* **2021**, *11*, 775. [[CrossRef](#)]
74. Diaconu, B.M. Recent advances and emerging directions in fire detection systems based on machine learning algorithms. *Fire* **2023**, *6*, 441. [[CrossRef](#)]
75. da Silva, J.C.; Silva, M.C.; Luz, E.J.; Delabrida, S.; Oliveira, R.A. Using mobile edge AI to detect and map diseases in citrus orchards. *Sensors* **2023**, *23*, 2165. [[CrossRef](#)]
76. Zhu, F.; Sun, Y.; Zhang, Y.; Zhang, W.; Qi, J. An improved MobileNetV3 mushroom quality classification model using images with complex backgrounds. *Agronomy* **2023**, *13*, 2924. [[CrossRef](#)]
77. Bandi, R.; Swamy, S.; Arvind, C.S. Leaf disease severity classification with explainable artificial intelligence using transformer networks. *Int. J. Adv. Technol. Eng. Explor.* **2023**, *10*, 278.
78. Buiu, C.; Dănilă, V.R.; Răduță, C.N. MobileNetV2 ensemble for cervical precancerous lesions classification. *Processes* **2020**, *8*, 595. [[CrossRef](#)]
79. Salim, F.; Saeed, F.; Basurra, S.; Qasem, S.N.; Al-Hadhrani, T. DenseNet-201 and Xception pre-trained deep learning models for fruit recognition. *Electronics* **2023**, *12*, 3132. [[CrossRef](#)]
80. Akter, S.; Shamrat, F.J.M.; Chakraborty, S.; Karim, A.; Azam, S. COVID-19 detection using deep learning algorithm on chest X-ray images. *Biology* **2021**, *10*, 1174. [[CrossRef](#)]
81. Mustak Un Nobi, M.; Rifat, M.; Mridha, M.F.; Alfarhood, S.; Safran, M.; Che, D. GLD-DET: Guava leaf disease detection in real-time using lightweight deep learning approach based on MobileNet. *Agronomy* **2023**, *13*, 2240. [[CrossRef](#)]
82. Ding, W.; Abdel-Basset, M.; Alrashdi, I.; Hawash, H. Next generation of computer vision for plant disease monitoring in precision agriculture: A contemporary survey, taxonomy, experiments, and future direction. *Inf. Sci.* **2024**, *665*, 120338. [[CrossRef](#)]
83. Saito, T.; Rehmsmeier, M. The precision-recall plot is more informative than the ROC plot when evaluating binary classifiers on imbalanced datasets. *PLoS ONE* **2015**, *10*, e0118432. [[CrossRef](#)]
84. Alzamil, Z.S.; Sait, A.R.W. A LeViT-EfficientNet-Based Feature Fusion Technique for Alzheimer’s Disease Diagnosis. *Appl. Sci.* **2024**, *14*, 3879. [[CrossRef](#)]
85. Manole, I.; Butacu, A.I.; Bejan, R.N.; Tiplica, G.S. Enhancing Dermatological Diagnostics with EfficientNet: A Deep Learning Approach. *Bioengineering* **2024**, *11*, 810. [[CrossRef](#)]
86. Maqsood, Y.; Usman, S.M.; Alhussein, M.; Aurangzeb, K.; Khalid, S.; Zubair, M. Model Agnostic Meta-Learning (MAML)-Based Ensemble Model for Accurate Detection of Wheat Diseases Using Vision Transformer and Graph Neural Networks. *Comput. Mater. Continua* **2024**, *79*, 2795–2811. [[CrossRef](#)]
87. Chen, P.; Dai, J.; Zhang, G.; Hou, W.; Mu, Z.; Cao, Y. Diagnosis of Cotton Nitrogen Nutrient Levels Using Ensemble MobileNetV2FC, ResNet101FC, and DenseNet121FC. *Agriculture* **2024**, *14*, 525. [[CrossRef](#)]
88. Akkajit, P.; Sukkuea, A.; Thongnonghin, B. Comparative analysis of five convolutional neural networks and transfer learning classification approach for microplastics in wastewater treatment plants. *Ecol. Inform.* **2023**, *78*, 102328. [[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.