scientific reports

Check for updates

OPEN

Explainable convolutional neural network architectures for high-performance taxonomic classification of gasteroid macrofungi

Eda Kumru[1], Fatih Ekinci[2], Abdullah Aydoğan[3], Mehmet Serdar Güzel[3], Mustafa Sevindik[4] & Ilgaz Akata[5]✉

Gasteroid fungi represent a morphologically diverse and taxonomically challenging group due to their convergent evolution and closed fruiting bodies. This study presents a novel deep learning-based framework for the classification of six macrofungi species: *Battarrea phalloides*, *Crucibulum laeve*, *Cyathus olla*, *C. striatus*, *Tulostoma brumale*, and *T. fimbriatum*. A total of 1200 high-resolution images were processed using eleven convolutional neural networks (CNNs), each pre-trained on ImageNet and fine-tuned for this classification task. DenseNet121 emerged as the best-performing model with 96.11% accuracy, 96.09% F1-score, and an AUC of 99.89%. ResNeXt and RepVGG followed closely with accuracies of 95.00% and 93.89%, respectively. Operational metrics showed that ShuffleNetV2 achieved the fastest inference time (0.80 s), while RepVGG exhibited the highest energy efficiency (16.5%). Explainable AI techniques, including Grad-CAM and Guided Backpropagation, were applied to enhance model interpretability, revealing biologically meaningful image regions. These results demonstrate that deep learning architectures can be effectively applied to fungal taxonomy with high accuracy and transparency. Furthermore, the proposed methodology is scalable and adaptable to other biological domains such as fungal spore classification, plant pollen analysis, and rare species identification in ecological monitoring. This study offers a robust, interpretable, and computationally efficient solution for automating biodiversity assessments using image-based AI techniques.

Keywords Deep learning, Gasteroid fungi, CNN classification, Explainable AI, Image-based taxonomy

Gasteroid fungi are a polyphyletic group within Basidiomycota, characterized by diverse shapes, enclosed fruiting bodies, and convergent features, which complicate their taxonomy. Identification of gasteroid fungi relies on both macroscopic and microscopic features, including fructification body shape, peridial layers, and gleba color, as well as spore size and ornamentation[1,2]. Molecular methods are essential for accurate classification and resolving phylogenetic relationships, as they overcome common morphological convergence[3–9].

This evolutionary shift, believed to be an adaptation to arid or fragmented environments, has led to the loss of active spore discharge mechanisms, replaced by passive dispersal through decay, rupture, or animal activity. Genera such as *Cyathus* Haller, *Crucibulum* Tul. and C. Tul., Battarrea, and *Tulostoma* Pers. within the order *Agaricales* Underw. exemplify this transformation[1–9].

These genera encompass ecologically diverse and morphologically distinctive species, underscoring the evolutionary and ecological breadth of gasteroid fungi. For instance, *Crucibulum laeve* (Huds.) Kambly and *Cyathus olla* (Batsch) Pers., commonly known as bird's nest fungi, are often found on woody debris and grasslands. At the same time, *Cyathus striatus* Willd. is notable for its hairy exterior and striate inner surface,

[1]Graduate School of Natural and Applied Sciences, Ankara University, 06830 Ankara, Turkey. [2]Institute of Artificial Intelligence, Ankara University, 06100 Ankara, Turkey. [3]Department of Computer Engineering, Faculty of Engineering, Ankara University, 06830 Ankara, Turkey. [4]Department of Biology, Faculty of Engineering and Natural Sciences, Osmaniye Korkut Ata University, Osmaniye, Turkey. [5]Department of Biology, Faculty of Science, Ankara University, 06100 Ankara, Turkey. ✉email: akata@science.ankara.edu.tr

thriving in temperate forest litter[2]. Among stalked puffballs, *Battarrea phalloides* (Dicks.) Pers. is of global conservation concern due to habitat loss, typically appearing in sandy, arid soils[8]. *Tulostoma brumale* Pers. is adapted to winter fruiting in dune systems, whereas *T. fimbriatum* Fr. is widely distributed in Europe, identifiable by its fringed pore and scaly stipe[9]. Though the morphological and ecological characteristics cited above are mainly taxonomically informative, these species are quite variable.

Despite their taxonomic and ecological importance, gasteroid fungi pose challenges for traditional classification due to morphological convergence and limited public awareness. To address this gap, the present study proposes a novel deep learning-based approach for accurately classifying six ecologically significant species: *Battarrea phalloides*, *Crucibulum laeve*, *Cyathus olla*, *C. striatus*, *Tulostoma brumale*, and *T. fimbriatum*. The goal is to develop a high-performance identification model that enhances fungal identification through photographs, while supporting conservation monitoring and public education in fungal biodiversity.

To achieve this, fungal image data were processed using convolutional neural networks (CNNs) for feature extraction and subsequently classified with a Kohonen self-organising map (SOM). This hybrid method is the first known attempt to integrate SOMs traditionally used for unsupervised clustering with CNN architectures in classifying gasteroid fungi[10]. Additionally, the study introduces Kolmogorov Arnold Network (KAN) layers into this framework and evaluates the performance of advanced architectures such as MaxViT-S, marking a pioneering step in the application of deep learning in fungal taxonomy[10,11]. The structure of this paper is as follows: Section "Materials and methods" outlines the dataset and experimental setup; Section "Results" presents the model evaluation and results; and Section "Discussion" discusses the limitations and future directions in AI-assisted fungal classification.

Materials and methods

A total of 1200 images representing six different mushroom species *Battarrea phalloides*, *Crucibulum laeve*, *Cyathus olla*, *C. striatus*, *Tulostoma brumale*, and *T. fimbriatum* (Fig. 1) were used in this study. The images were sourced from the authors' own dataset and the Global Core Biodata Resource (www.gbif.org)[12], with the specific proportions of data from each source detailed in Table 1. All images were acquired at a fixed resolution of 300 dpi and in JPEG format (Table 1).

The complete dataset was partitioned as follows: 65% for training, 15% for validation, 15% for testing, and 5% reserved for fine-tuning. This structured split ensured that the model development process, hyperparameter tuning, and final evaluation were conducted on independent subsets. This dataset can be accessed at https://tinyurl.com/ye5utx2h and https://tinyurl.com/yxj5xh85.

To enhance robustness, generalization, and mitigate overfitting, data augmentation techniques were applied only to the training subset. Each image was subjected to several transformations[13]. These included random rotations to simulate different viewing angles and horizontal flipping to mimic mirror-image scenarios[14]. Brightness and contrast adjustments were used to address lighting variability. Gaussian blur was applied to introduce slight distortions[15]. Finally, random cropping combined with resizing helped enforce scale invariance. These augmentation procedures effectively tripled the number of training images, thereby enriching the dataset without compromising its integrity[16].

Eleven different deep learning (DL) models were employed for classification, including DenseNet121, EfficientNetB0, EfficientNetB4, GhostNet, RegNetX-400MF, RegNetX-800MF, RepVGG, ResNeXt, ShuffleNetV2, SqueezeNet, and Xception. Each model was initialized with pre-trained weights on the ImageNet dataset. In this study, PyTorch[17] was employed as the primary deep learning framework. Model architectures were loaded using both the official TorchVision library[18] and the timm (PyTorch Image Models) package[19].Through TorchVision, models such as DenseNet121, EfficientNetB0, EfficientNetB4, RegNetX-400MF, RegNetX-800MF, ResNeXt, ShuffleNetV2, and SqueezeNet were utilized, whereas GhostNet, RepVGG, and Xception were integrated via the timm library. For data preprocessing, TorchVision's transforms and datasets modules were applied. Mathematical operations were supported by NumPy[20] while additional evaluation metrics, such as log loss, were computed using scikit-learn[21]. System performance monitoring was conducted with psutil[22], and standard Python libraries such as os, time, and json were employed for file management and execution control[23].

DenseNet121 introduces a dense connectivity pattern where each layer is directly connected to every subsequent layer, promoting feature reuse throughout the network[24]. DenseNet121 minimises the vanishing gradient problem and model redundancy by using dense blocks and transition layers that downsample feature maps and manage complexity. This design ensures efficient information flow and strong feature propagation, making it effective for diverse deep learning tasks[25]. EfficientNetB0 uses compound scaling to balance depth, width, and resolution, achieving high accuracy with low computational cost, ideal for image classification[24,25].

EfficientNetB0 employs a compound scaling method that uniformly adjusts network depth, width, and input resolution using fixed scaling coefficients[26]. It uses mobile inverted bottleneck blocks and squeeze-and-excitation modules for better feature recalibration, with balanced scaling optimising all dimensions to enhance model efficiency[27]. EfficientNetB0 effectively allocates computational resources, offering a scalable and streamlined architecture for deep learning applications[26,27].

EfficientNetB4 builds upon the compound scaling strategy by proportionally increasing network depth, width, and input resolution compared to its counterparts[28]. Its architecture utilizes mobile inverted bottleneck convolution blocks along with squeeze-and-excitation modules, which serve to refine and recalibrate feature representations[29]. EfficientNetB4's systematic scaling balances expansion across all dimensions, optimising parameters and efficiency while maintaining a streamlined structure for deep learning[28,29].

GhostNet is characterized by its innovative ghost module design, which generates intrinsic feature maps through a series of inexpensive linear operations[30]. This architecture effectively reduces computational overhead while maintaining the integrity of feature representation[31]. GhostNet emphasises generating redundant



Fig. 1. This figure shows six macrofungi species from the dataset, illustrating their varied textures, structures, and colours. The species include: (**a**) *Battarrea phalloides*, (**b**) *Crucibulum laeve*, (**c**) *Cyathus olla*, (**d**) *Cyathus striatus*, (**e**) *Tulostoma brumale* and (**f**) *Tulostoma fimbriatum*.

Mushroom species name	Captured by the authors (%)	Open-source resources (%)	Type of photograph	Resolution
Battarrea phalloides	< 25	> 75	JPEG	300 dpi
Crucibulum laeve	< 21	> 79	JPEG	300 dpi
Cyathus olla	< 20	> 80	JPEG	300 dpi
Cyathus striatus	< 23	> 77	JPEG	300 dpi
Tulostoma brumale	< 19	> 81	JPEG	300 dpi
Tulostoma fimbriatum	< 24	> 76	JPEG	300 dpi

Table 1. Overview of the six mushroom species included in the dataset, showing the proportion of images, photograph format and resolution.

yet informative features to enhance resource efficiency. Its modular structure also supports scalability and streamlined construction, making it adaptable to a wide range of deep learning applications[30,31].

RegNetX-400MF is structured around a simple and scalable design paradigm that employs stage-wise blocks with a consistent parameterization scheme. The network uses sequential bottleneck blocks to transform

features and support gradient flow, with a design optimised for 400 million FLOPs to balance complexity and performance. The model's straightforward design enhances interpretability and provides a clear framework for analyzing resource allocation across layers[32].

RegNetX-800MF is designed with a scalable architecture that employs stage-wise bottleneck blocks to facilitate effective feature transformation and gradient propagation[33]. The model is carefully structured to maintain a balance between network capacity and computational cost within an 800 million FLOPs budget. Organised into sequential stages, it refines feature representations through systematically designed layers. This modular and straightforward design enhances resource efficiency while supporting clear performance analysis at each stage[32,33].

RepVGG employs a VGG-inspired architecture that utilizes structural re-parameterization to transition from a multi-branch training configuration to a streamlined, single-branch model during inference[34,35]. RepVGG balances complexity and efficiency by using a simple structure of convolutional layers with batch normalization and activations, benefiting from higher capacity during training and efficient computation in deployment through re-parameterization[35,36].

ResNeXt builds upon the principles of residual learning by incorporating a split-transform-merge strategy that aggregates parallel transformations[37]. ResNeXt introduces cardinality by employing parallel group convolutions to capture diverse features, enhancing representational capacity without simply increasing depth or width. This modular and efficient design supports balanced feature extraction and residual mapping, providing a scalable and effective framework for deep learning models[37,38].

ShuffleNet-V2 is engineered with a focus on optimizing computational efficiency through innovative channel split and shuffle operations[39]. ShuffleNet-V2 uses pointwise group convolutions and channel shuffle to boost inter-channel communication while reducing memory cost. Its design ensures balanced resource use and efficient feature transformation, offering a scalable and computationally efficient framework[39,40].

SqueezeNet is characterized by its compact architecture, primarily achieved through the innovative use of fire modules[41]. Each fire module uses a 1×1 squeeze layer to reduce dimensionality, followed by 1×1 and 3×3 expand layers to restore spatial details. This design lowers parameter count while preserving feature quality, enabling efficient computation and memory use ideal for resource-constrained environments[41,42].

Xception is built upon the concept of depthwise separable convolutions, which decouple spatial and cross-channel correlations to enhance computational efficiency[43]. Xception replaces traditional inception modules with a streamlined design that supports deeper networks without increasing parameters. Residual connections aid gradient flow and feature extraction, while factorized convolutions balance efficiency and representational power[43,44].

The model was trained on the custom dataset with a batch size of 64 and an initial learning rate of 0.0001 for 10 epochs, using CrossEntropyLoss and the Adam optimizer. After this initial training, a fine-tuning phase was performed on a reserved subset to improve performance. During fine-tuning, the batch size remained 64, but the learning rate was reduced to 0.00005 for more controlled updates. This phase also lasted 10 epochs with the same loss function and optimizer. Using consistent hyperparameters across all models ensured fair comparison, stable convergence, and efficient multi-class classification.

To assess the performance of each CNN model, 12 distinct metrics were employed to capture various aspects of classification outcomes, including correctness, reliability, and resource usage[10,11]. These metrics comprise accuracy, precision, recall, F1-score, specificity, log loss, Matthews correlation coefficient (MCC), confusion matrix, ROC & AUC, inference time, memory usage, and energy efficiency[41]. Together, they offer a comprehensive view of how each model handles both predictive and operational considerations. For clarity, the mathematical formulations for seven of these metrics are provided below[11,45]. These equations detail how accuracy, precision, recall, F1-score, specificity, log loss, and the Matthews correlation coefficient (MCC) are calculated:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{1}$$

$$Precision = \frac{TP}{TN + FP} \tag{2}$$

$$Recall = \frac{TP}{TP + FN} \tag{3}$$

$$F1{-}Score = \frac{2 \times (Precision \times Recall)}{Precision + Recall} \tag{4}$$

$$Specificity = \frac{TN}{TN + FP} \tag{5}$$

$$Log\ Loss = -\frac{1}{N} \sum_{i=1}^{N} \sum_{j=1}^{M} y_{ij} \log\left(\widehat{y}ij\right) \tag{6}$$

$$Matthews\ Correlation\ Coefficient = \frac{(TP \times TN) - (FP \times FN)}{\sqrt{(TP + FP) \times (TP + FN) \times (TN + FP) \times (TN + FN)}} \tag{7}$$

In the equations, TP (true positive) represents the number of positive instances correctly classified as positive (Eqs. 1, 2, 3 and 7), whereas TN (true negative) denotes the number of negative instances correctly classified

as negative (Eqs. 1, 2, 5 and 7). Conversely, FP (false positive) is the number of negative instances incorrectly classified as positive (Eqs. 1, 2, 5 and 7), and FN (false negative) is the number of positive instances incorrectly classified as negative (Eqs. 1, 3 and 7). The term y_{ij} refers to an indicator variable for sample i, which takes a value of 1 if that sample belongs to class j and 0 otherwise; N denotes the total number of samples in the dataset, and M indicates the total number of classes considered in the classification task (Eq. 6) [11,45].

Accuracy measures the overall proportion of correct predictions (1), while precision quantifies the fraction of true positive predictions among all positive predictions (2), and recall (sensitivity) evaluates the proportion of actual positives correctly identified (3). The F1-score, being the harmonic mean of precision and recall, balances these two aspects (4), whereas specificity indicates the proportion of true negatives accurately classified (5). Log loss (cross-entropy loss) assesses the quality of predicted probabilities by penalizing overconfident errors (6). Logloss is a metric used to evaluate the accuracy of a model's predictions in classification problems. This metric measures the alignment between the probabilities predicted by the model and the actual class labels. For each instance, the logarithmic difference between the predicted probability and the actual label is calculated. When higher probabilities are assigned to the correct classes, the logloss value decreases, while it increases when higher probabilities are assigned to incorrect classes. Thus, logloss serves as a metric that measures the model's accuracy and reliability. And the Matthews correlation coefficient (MCC) offers a balanced measure that takes into account true and false predictions across both classes (7). Additionally, the confusion matrix offers a detailed view of true and false classifications, while the ROC curve and AUC (Area Under the Curve) highlight the balance between sensitivity and specificity across thresholds. Operational metrics like inference time, memory usage, and energy efficiency assess the model's speed, resource needs, and power consumption during deployment[11,45].

A confusion matrix summarises a classification model's performance by comparing actual and predicted labels. It shows true positives, true negatives, false positives, and false negatives, allowing a detailed assessment of accuracy and error types, and highlighting where misclassifications occur.This confusion matrix can be access at https://tinyurl.com/yxj5xh85.

In order to thoroughly evaluate classification performance, a set of metrics (1–7) was calculated for each model examined. Given the inherently opaque nature of deep neural networks often referred to as 'black boxes' three Explainable Artificial Intelligence (XAI) techniques were employed to enhance interpretability: Grad-CAM, thresholded Grad-CAM, and Guided Backpropagation. Gradient-weighted Class Activation Mapping (Grad-CAM) generates coarse heatmaps corresponding to the target class by propagating class-specific gradients into the final convolutional layer[46]. To further refine these heatmaps, a threshold of 100 was applied to the pixel intensity values, thereby generating binary masks that isolate the most salient areas above this threshold level. Guided Backpropagation improves standard backpropagation by passing only positive gradients, producing high-resolution saliency maps[47]. Together, these techniques highlight both coarse and fine-grained image features influencing model predictions, thus clarifying model behaviour and strengthening trust in their use for macrofungi classification[46,47].

Results

In this section, a comprehensive comparison of the CNN models is presented from multiple perspectives. This study aims to assess both the predictive performance and the practical applicability of the models in macrofungi classification. Initially, graphs illustrating training and fine-tuning accuracy are provided to examine convergence behavior and stability during model optimization. Subsequently, test results are compared using accuracy and F1-score graphs to evaluate the effectiveness and balance of predictions. ROC curves are also plotted to highlight each model's discrimination capacity based on test data. Inference time is compared to assess computational efficiency during deployment, while memory usage and energy efficiency metrics are analyzed to gauge operational demands. In addition, Grad-CAM and Guided Backpropagation visualizations are employed to elucidate the models' decision-making processes by highlighting the salient features contributing to their predictions. Finally, a detailed table summarizing all test metrics is provided to offer a holistic view of performance, thereby enabling a robust and multi-faceted evaluation of the models' capabilities.

In the graphs (Figs. 2, 3 and 4), which depict a 20-epoch cycle with an initial training phase of 10 epochs followed by a fine-tuning phase of 10 epochs, distinct performance levels are observed across all models. DenseNet121, ResNeXt, Xception, and EfficientNetB4 rapidly converge early and maintain high accuracy throughout fine-tuning, reflecting robust feature extraction and effective optimization. In contrast, RegNetX-400MF shows a slower, more gradual improvement, indicating a relatively limited adaptation to the dataset. EfficientNetB0, RegNetX-800MF, and RepVGG exhibit steady, moderate gains over both phases, achieving intermediate accuracy levels by the end of fine-tuning. Meanwhile, ShuffleNetV2, SqueezeNet, and GhostNet display more modest improvements, with their accuracy curves plateauing at lower levels compared to the top-performing models. These observations, based solely on the training and fine-tuning data, clearly reveal differences in the models' ability to adapt to the dataset under the applied two-stage training strategy.

In this bar chart (Fig. 5), the accuracy and F1-scores of the evaluated CNN architectures are compared. DenseNet121 achieves the strongest results with 96.11% accuracy and a 96.09% F1-score, clearly outperforming the others and confirming its superior classification capability, while ResNeXt follows closely with 95.00% and 95.00%, and RepVGG (93.89% / 93.92%) together with EfficientNetB4 (93.33% / 93.36%) also demonstrate robust generalization and reliable predictive power. In the mid-range, SqueezeNet achieves 91.11% accuracy and 91.12% F1-score, and Xception reports 90.56% and 90.55%, both indicating solid but not top-tier performance, whereas EfficientNetB0 (87.22% / 87.04%), ShuffleNetV2 (87.22% / 87.21%), and GhostNet (87.78% / 87.78%) remain competitive though clearly below the leading models. RegNetX-800MF, at 85.56% accuracy and 85.43% F1-score, performs modestly, while RegNetX-400MF ranks the lowest with 79.44% and 79.15%, underscoring its weaker representational capacity. Beyond accuracy and F1-scores, the precision and recall values confirm the stability of these models: DenseNet121 achieves 96.20% precision and 96.11% recall, yielding a balanced ratio of

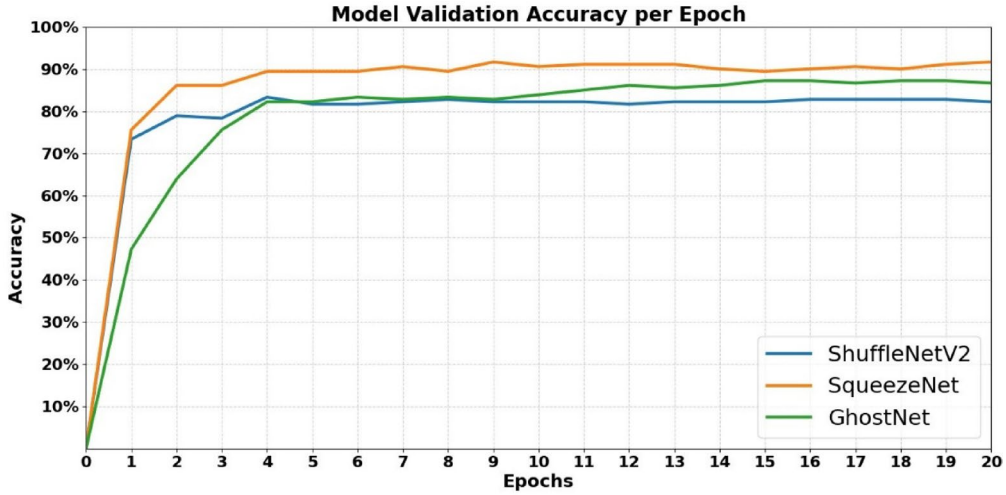

Fig. 2. Accuracy curves over 20 epochs including 10 epochs of training followed by 10 epochs of fine-tuning for deep learning models ShuffleNetV2, SqueezeNet, and GhostNet.

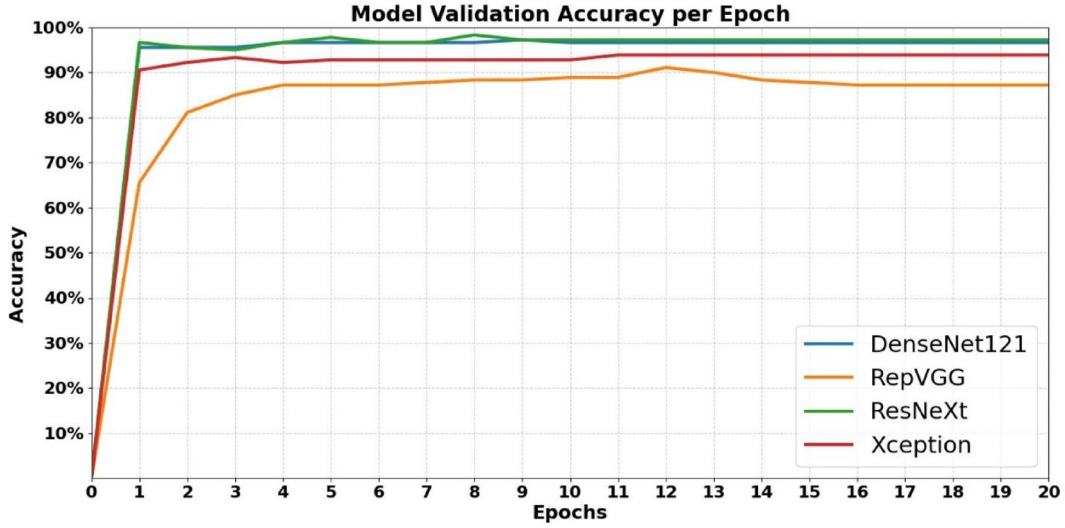

Fig. 3. Accuracy curves over 20 epochs including 10 epochs of training followed by 10 epochs of fine-tuning for deep learning models DenseNet121, RepVGG, ResNeXt, and Xception.

about 99.9%, and similar equilibrium is seen in ResNeXt (95.08% / 95.00%), RepVGG (94.04% / 93.89%), and EfficientNetB4 (93.63% / 93.33%), all of which maintain strong precision–recall alignment that supports their high ranking. Even models with lower accuracy, such as GhostNet (87.92% / 87.78%), ShuffleNetV2 (88.02% / 87.22%), and SqueezeNet (91.46% / 91.11%), preserve balanced ratios above 99%, showing that their limitations stem more from architectural capacity than instability in predictive behavior. Overall, these results highlight that while absolute performance levels vary across architectures, precision–recall balance remains consistently strong, reinforcing the robustness of the evaluated models.

In the ROC curves (Fig. 6), most models demonstrate high discriminative power, as indicated by their AUC values. ResNeXt and DenseNet121 emerge with the highest AUC scores, suggesting near-perfect distinction between classes. EfficientNetB4 and Xception follow closely, reflecting similarly strong predictive capabilities. RepVGG and SqueezeNet also achieve robust AUC values, indicating effective class separation. Meanwhile, EfficientNetB0, RegNetX-800MF, GhostNet, and ShuffleNetV2 exhibit slightly lower but still commendable AUC scores, highlighting consistent, albeit less dominant, performance. RegNetX-400MF concludes the ranking with the lowest AUC among the evaluated models, yet it remains within a competitive range overall. These observations underscore the varying degrees to which each architecture excels at differentiating macrofungi species, providing valuable insights into their underlying capacity for accurate classification.

In the bar chart comparing inference times (Fig. 7), ShuffleNetV2 achieves the fastest performance with an inference time of approximately 0.80 s, while Xception records the slowest at around 10.08 s. Several models

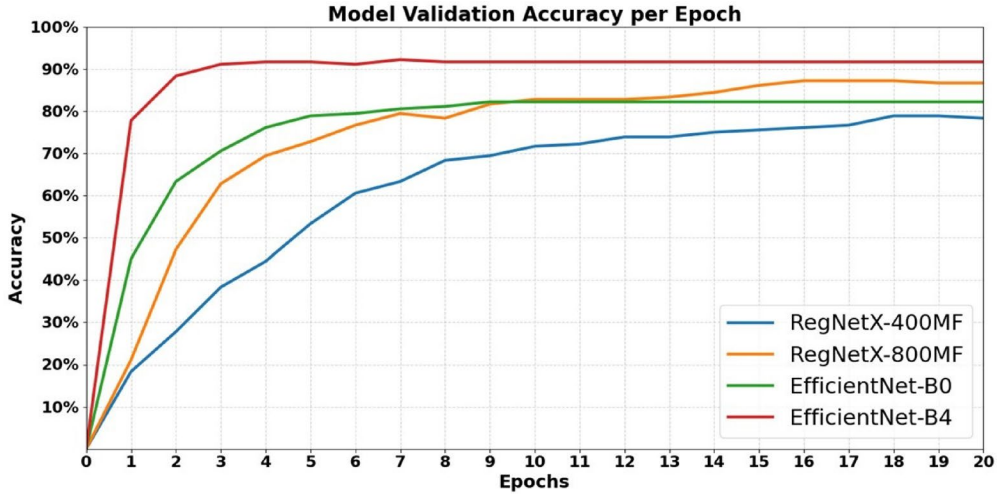

Fig. 4. Accuracy curves over 20 epochs including 10 epochs of training followed by 10 epochs of fine-tuning for deep learning models RegNetX-400MF, RegNetX-800MF, EfficientNetB0, and EfficientNetB4.

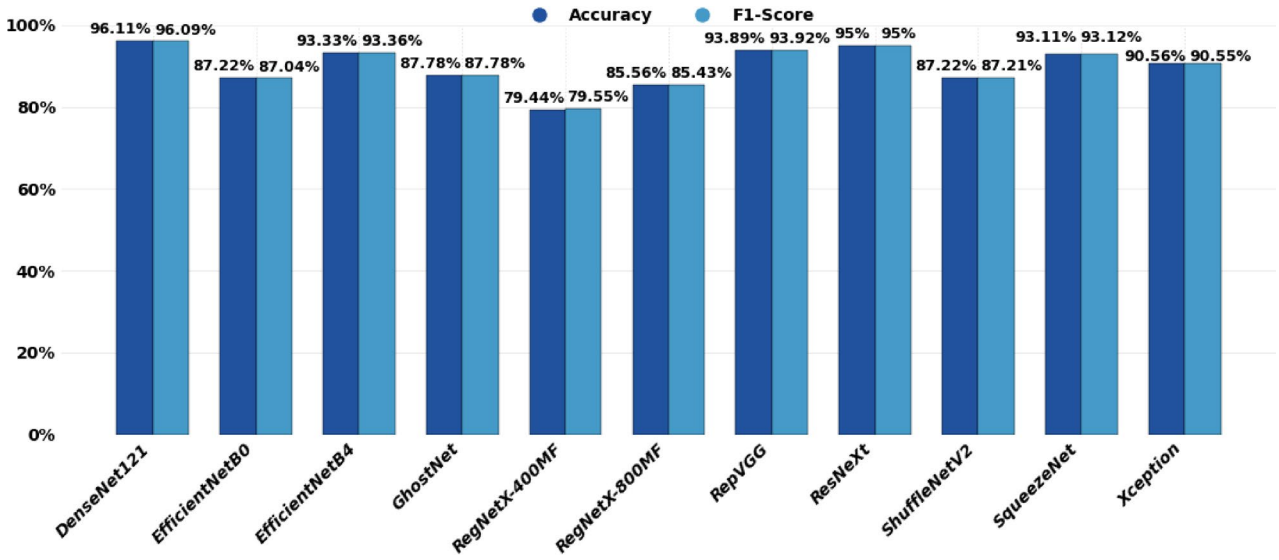

Fig. 5. Accuracy and F1-score values for deep learning models.

fall into an intermediate range: DenseNet121 requires about 5.29 s, EfficientNetB4 takes roughly 8.16 s, and ResNeXt reaches 9.67 s. GhostNet, RegNetX-400MF, and SqueezeNet remain below 2 s, indicating highly efficient architectures. RepVGG (3.58 s) and EfficientNetB0 (3.08 s) occupy a moderate interval, balancing computational load with representational power. These values represent the time required to process a batch of images, reflecting the models' efficiency under grouped input conditions. It should also be noted that inference time is influenced by both the selected batch size and the underlying hardware configuration, which together determine the overall computational performance.

In the bar chart (Fig. 8) comparing memory usage, most models operate within a narrow memory range of approximately 55–57%. SqueezeNet stands out at 55.2%, reflecting the lowest memory consumption, whereas DenseNet121 peaks at 57.5%. In the bar chart (Fig. 9) comparing energy efficiency, however, varies more dramatically. RepVGG leads at 16.5%, followed by RegNetX-400MF at 12.8%, DenseNet121 at 11.4%, and EfficientNetB4 at 11.2%. A larger group ShuffleNetV2 at 9.7%, EfficientNetB0 at 9.3%, RegNetX-800MF at 8.6%, ResNeXt at 8.5%, Xception at 8.2%, GhostNet at 8.1%, and SqueezeNet at 8.0% clusters below 10%. These differences highlight how architectural choices can significantly influence both resource consumption and sustainability, underscoring the importance of selecting models that align with specific operational constraints.

DenseNet121 achieves an accuracy of 96.11%, precision of 96.20%, recall of 96.11%, and an F1-score of 96.09%, reflecting a highly balanced classification performance. The model attains an AUC of 99.89% and a specificity of 99.22%, indicating near-perfect discrimination and a low rate of false positives. Its log loss of %9.11 suggests well-calibrated probabilistic outputs, while an MCC of 95.36% highlights the strong correlation between

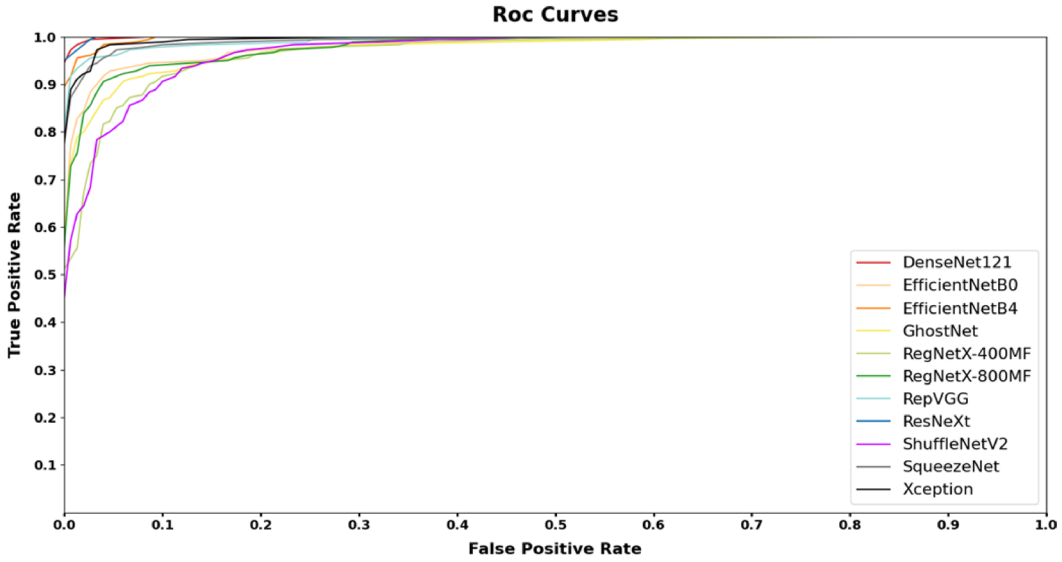

Fig. 6. ROC curves for deep learning models.

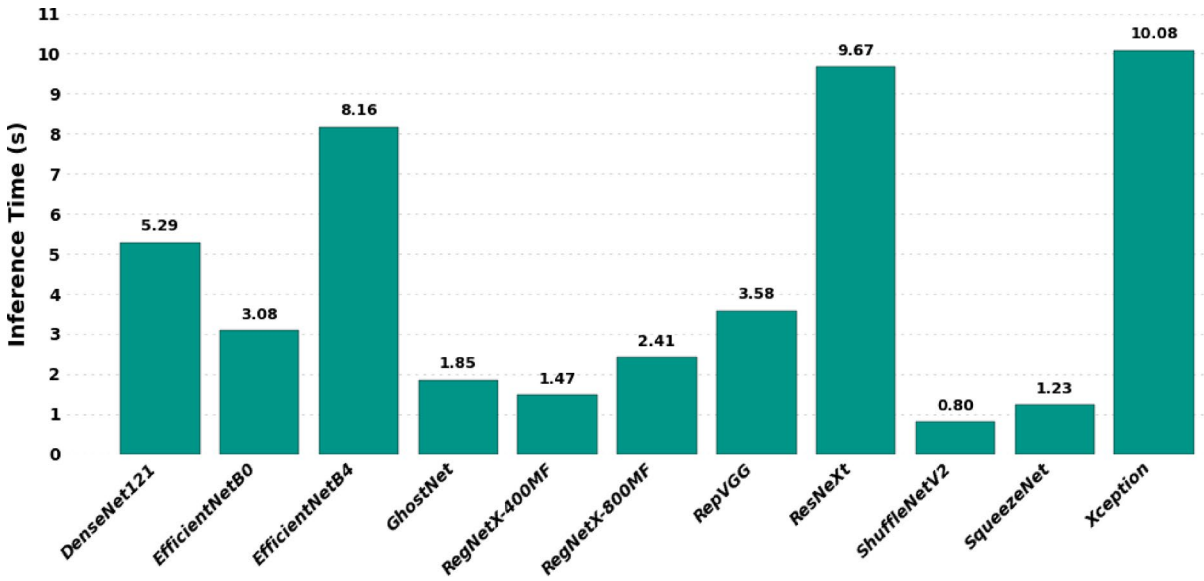

Fig. 7. Inference times for deep learning models.

predictions and true labels. Collectively, these results underscore DenseNet121's robust feature representation and effective decision-making for macrofungi classification. EfficientNetB0 yields an accuracy of 87.22%, precision of 87.20%, recall of 87.22%, and an F1-score of 87.04%, indicating a balanced capacity for detecting both positive and negative classes. The model achieves an AUC of 97.66% and a specificity of 97.44%, demonstrating robust discrimination against false positives. Its log loss of 71.60% suggests that while the predicted probabilities are reasonably aligned with actual outcomes, there remains room for calibration improvements. Meanwhile, an MCC of 84.74% signifies a solid correlation between predicted and true labels, affirming EfficientNetB0's overall reliability. EfficientNetB4 achieves an accuracy of 93.33%, precision of 93.63%, recall of 93.33%, and an F1-score of 93.36%, indicating robust classification performance with a well-balanced detection of both positive and negative cases. Its AUC of 99.69% and specificity of 98.67% demonstrate a strong ability to distinguish between classes and limit false positives. The model's log loss of 16.96% reflects well-calibrated probabilistic outputs, while an MCC of 92.04% further confirms a high correlation between predictions and true labels (Table 2).

GhostNet achieves an accuracy of 87.78%, precision of 87.92%, recall of 87.78%, and an F1-score of 87.78%, reflecting a balanced classification performance. The model attains an AUC of 97.19% and a specificity of 97.56%, indicating robust discriminative ability and a low incidence of false positives. Its log loss of 87.39% suggests moderate calibration of its probabilistic outputs, while an MCC of 85.36% highlights a satisfactory correlation between predictions and true labels. Collectively, these metrics underscore GhostNet's effective feature extraction

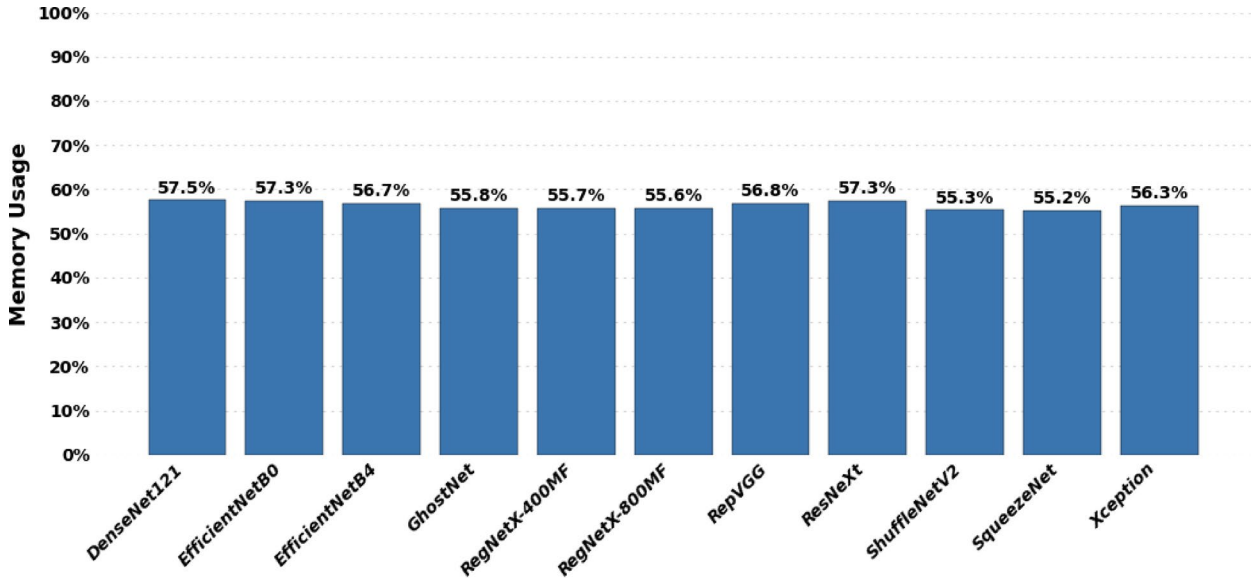

Fig. 8. Memory usage for deep learning models.

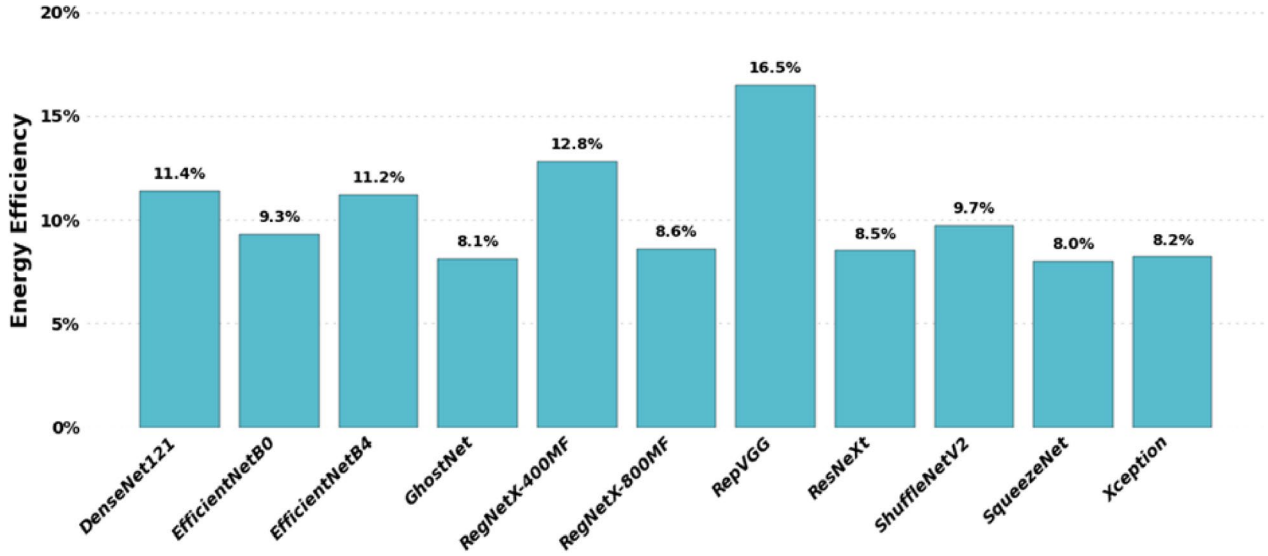

Fig. 9. Energy efficiency for deep learning models.

and reliable decision-making for macrofungi classification. RegNetX-400MF achieves an accuracy of 79.44%, precision of 79.51%, recall of 79.44%, and an F1-score of 79.15%, reflecting a moderate classification performance with balanced detection of both positive and negative cases. The model attains an AUC of 96.36% and a specificity of 95.89%, indicating a strong capability to discriminate between classes. Its log loss of 88.86% suggests that the probabilistic outputs are less well-calibrated compared to higher-performing models, while an MCC of 75.46% highlights a moderate correlation between predictions and true labels. Collectively, these metrics underscore RegNetX-400MF's capacity for feature representation in macrofungi classification, while also pointing to areas for potential calibration improvement. RegNetX-800MF achieves an accuracy of 85.56%, precision of 85.72%, recall of 85.56%, and an F1-score of 85.43%, reflecting a moderate classification performance with balanced detection of both positive and negative cases. The model attains an AUC of 97.64% and a specificity of 97.11%, indicating robust discriminative ability and a low incidence of false positives. Its log loss of 72.78% suggests that the probabilistic outputs are less well-calibrated compared to higher-performing models, while an MCC of 82.74% highlights a moderate correlation between predictions and true labels. Collectively, these metrics underscore RegNetX-800MF's capacity for effective feature representation in macrofungi classification while also indicating potential areas for calibration improvement. RepVGG achieves an accuracy of 93.89%, precision of 94.04%, recall of 93.89%, and an F1-score of 93.92%, reflecting strong classification performance with balanced detection of both positive and negative cases. The model attains an AUC of 99.15% and a specificity of 98.78%,

Models	Accuracy	Precision	Recall	F1-Score	Specificity	AUC	MCC	Log Loss
DenseNet121	0.9611	0.9620	0.9611	0.9609	0.9922	0.9989	0.9536	0.0911
EfficientNetB0	0.8722	0.8720	0.8722	0.8704	0.9744	0.9766	0.8474	0.7160
EfficientNetB4	**0.9333**	**0.9363**	**0.9333**	**0.9336**	**0.9867**	**0.9969**	**0.9204**	**0.1696**
GhostNet	0.8778	0.8792	0.8778	0.8778	0.9756	0.9719	0.8536	0.8739
RegNetX-400MF	0.7944	0.7951	0.7944	0.7915	0.9589	0.9636	0.7546	0.8886
RegNetX-800MF	0.8556	0.8572	0.8556	0.8543	0.9711	0.9764	0.8274	0.7278
RepVGG	**0.9389**	**0.9404**	**0.9389**	**0.9392**	**0.9878**	**0.9915**	**0.9268**	**0.3081**
ResNeXt	**0.9500**	**0.9508**	**0.9500**	**0.9500**	**0.9900**	**0.9990**	**0.9402**	**0.0937**
ShuffleNetV2	0.8722	0.8802	0.8722	0.8721	0.9744	0.9663	0.8482	1.4142
SqueezeNet	0.9111	0.9146	0.9111	0.9112	0.9822	0.9900	0.8940	0.4016
Xception	0.9056	0.9081	0.9056	0.9055	0.9811	0.9936	0.8871	0.2506

Table 2. Performance comparison of deep learning models based on accuracy, precision, recall, F1-score, specificity, AUC score, MCC (Matthews correlation coefficient) score and log loss. Significant values are in bold.

indicating robust discriminative ability and a low incidence of false positives. Its log loss of 30.81% suggests well-calibrated probabilistic outputs, while an MCC of 92.68% highlights a strong correlation between predictions and true labels. Collectively, these metrics underscore RepVGG's effective feature representation and reliable decision-making for macrofungi classification (Table 2).

ResNeXt achieves an accuracy of 95.00%, precision of 95.08%, recall of 95.00%, and an F1-score of 95.00%, reflecting robust classification performance with balanced detection of both positive and negative cases. The model attains an AUC of 99.90% and a specificity of 99.00%, indicating near-perfect discriminative ability and a very low rate of false positives. Its log loss of 9.37% suggests well-calibrated probabilistic outputs, while an MCC of 94.02% highlights a strong correlation between predictions and true labels. Collectively, these metrics underscore ResNeXt's effective feature representation and reliable decision-making for macrofungi classification. ShuffleNetV2 achieves an accuracy of 87.22%, precision of 88.02%, recall of 87.22%, and an F1-score of 87.21%, reflecting balanced classification performance with consistent detection of both positive and negative cases. The model attains an AUC of 96.63% and a specificity of 97.44%, indicating strong discriminative ability and a low rate of false positives. Its log loss of 141.42% suggests that the probabilistic outputs are less well-calibrated compared to higher-performing models, while an MCC of 84.82% highlights a moderate correlation between predictions and true labels. Collectively, these metrics underscore ShuffleNetV2's capacity for effective feature representation and decision-making in macrofungi classification, while also indicating potential areas for calibration improvement. SqueezeNet achieves an accuracy of 91.11%, precision of 91.46%, recall of 91.11%, and an F1-score of 91.12%, reflecting robust classification performance with balanced detection of both positive and negative cases. The model attains an AUC of 99.00% and a specificity of 98.22%, indicating strong discriminative capability and a low incidence of false positives. Its log loss of 40.16% suggests that the probabilistic outputs are reasonably well-calibrated, while an MCC of 89.40% highlights a strong correlation between predictions and true labels. Collectively, these metrics underscore SqueezeNet's effective feature representation and reliable decision-making for macrofungi classification. Xception achieves an accuracy of 90.56%, precision of 90.81%, recall of 90.56%, and an F1-score of 90.55%, reflecting robust classification performance with balanced detection of both positive and negative cases. The model attains an AUC of 99.36% and a specificity of 98.11%, indicating strong discriminative capability and a low incidence of false positives. Its log loss of 25.06% suggests that the probabilistic outputs are well-calibrated, while an MCC of 88.71% highlights a strong correlation between predictions and true labels. Collectively, these metrics underscore Xception's effective feature representation and reliable decision-making for macrofungi classification (Table 2).

The predictive performance of each model was evaluated using a comprehensive set of metrics, including Accuracy, Precision, Recall, F1-Score, Specificity, AUC Score, MCC Score, and Log Loss. Among the evaluated architectures, some models exhibit notably superior performance, while others display more modest classification capabilities. DenseNet121, ResNeXt, and RepVGG consistently stand out with highly balanced feature representation and exceptional discrimination between classes. Xception, EfficientNetB4, and SqueezeNet also perform strongly, demonstrating robust predictive power and reliable calibration of probabilistic outputs. In contrast, EfficientNetB0 and RegNetX-800MF offer solid yet moderately lower performance, indicating a well-rounded but less dominant classification capacity. GhostNet, ShuffleNetV2, and RegNetX-400MF tend to achieve more modest results, suggesting that their feature extraction and calibration might benefit from further refinement. Overall, these differences highlight distinct levels of success in accurately distinguishing macrofungi species, reflecting the trade-offs inherent in each architectural design. The models with superior outcomes tend to provide more reliable and consistent predictions, whereas those with lower performance indicate areas for potential improvement. Collectively, this comparative analysis underscores the importance of aligning model selection with the specific demands of the classification task and contributes to a comprehensive understanding of the relative strengths and limitations of each deep learning approach (Table 2).

DenseNet121 exhibits a memory usage of 57.50%, reflecting moderate consumption of system resources. Its energy efficiency stands at 11.40%, indicating a relatively balanced power utilization profile. Additionally, DenseNet121's average inference time of 5.28 s underscores a moderate processing speed that aligns with its

robust classification performance. EfficientNetB0 utilizes 57.30% of memory, indicating moderate resource consumption. Its energy efficiency is 9.30%, suggesting a balanced power usage profile. Furthermore, the model's average inference time of 3.08 s underscores relatively swift processing compared to more computationally intensive architectures. EfficientNetB4 consumes 56.70% of memory, reflecting moderate resource usage relative to its capacity. Its energy efficiency is 11.20%, suggesting a balanced power consumption profile. Additionally, the model's average inference time of 8.16 s indicates a relatively higher computational demand compared to lighter architectures (Table 3).

GhostNet consumes 55.80% of memory, indicating efficient resource utilization. Its energy efficiency is 8.10%, suggesting a favorable power consumption profile. Additionally, the model's average inference time of 1.85 s highlights its rapid processing capabilities relative to more complex architectures. RegNetX-400MF consumes 55.70% of memory, indicating efficient resource utilization. Its energy efficiency is 12.80%, suggesting a relatively higher power consumption profile. Additionally, the model's average inference time of 1.46 s highlights its rapid processing capabilities, making it suitable for scenarios where computational speed is critical. RegNetX-800MF consumes 55.60% of memory, indicating efficient resource utilization. Its energy efficiency is 8.60%, suggesting a favorable power consumption profile. Additionally, the model's average inference time of 2.41 s highlights its rapid processing capabilities, making it well-suited for scenarios that demand timely inference. RepVGG consumes 56.80% of memory, indicating efficient resource utilization. Its energy efficiency is 16.50%, suggesting comparatively higher power consumption. Additionally, the model's average inference time of 3.57 s highlights a balanced processing capability, making it a viable option for applications that require both robust performance and reasonable computational speed (Table 3).

ResNeXt consumes 57.30% of memory, indicating efficient resource utilization. Its energy efficiency is 8.50%, suggesting a favorable power consumption profile. Additionally, the model's average inference time of 9.67 s highlights its relatively higher computational demand, making it well-suited for scenarios where comprehensive processing is prioritized. ShuffleNetV2 consumes 55.30% of memory, indicating efficient resource utilization. Its energy efficiency is 9.70%, suggesting a favorable power consumption profile. Additionally, the model's average inference time of 0.80 s highlights its exceptionally rapid processing capabilities, making it an excellent choice for applications requiring real-time inference. SqueezeNet consumes 55.20% of memory, indicating efficient resource utilization. Its energy efficiency is 8.00%, suggesting a favorable power consumption profile. Additionally, the model's average inference time of 1.22 s highlights its swift processing capabilities, making it well-suited for applications that require both effective performance and rapid inference. Xception consumes 56.30% of memory, indicating efficient resource utilization. Its energy efficiency is 8.20%, suggesting a favorable power consumption profile. Additionally, the model's average inference time of 10.07 s highlights its relatively higher computational demand, making it well-suited for scenarios where comprehensive processing is prioritized (Table 3).

The operational efficiency of each model was assessed based on average inference time, memory usage, and energy efficiency. Among the evaluated models, distinct differences in operational efficiency are evident. ShuffleNetV2 stands out with exceptional processing speed and minimal resource demands, making it ideal for time-critical applications. SqueezeNet follows closely, offering rapid inference combined with efficient memory and power usage. GhostNet also demonstrates strong operational performance with fast processing and competitive resource utilization. RegNetX-400MF and RegNetX-800MF deliver commendable speed and resource efficiency, although their energy profiles differ slightly. EfficientNetB0 and EfficientNetB4 exhibit balanced performance, with moderate inference speeds and stable resource consumption. RepVGG achieves acceptable processing times but is marked by higher energy consumption relative to its peers. Finally, DenseNet121, ResNeXt, and Xception display slower inference speeds, which may limit their suitability for real-time scenarios despite robust overall resource management. Collectively, these observations highlight that selecting a model for practical applications requires careful consideration of both computational efficiency and operational constraints (Table 3).

In the comprehensive evaluation of both predictive performance and operational efficiency, distinct tiers emerge among the models. DenseNet121, ResNeXt, and RepVGG consistently deliver exceptional classification

Models	Memory Usage	Energy Efficiency	Inference Time (s)
DenseNet121	0.5750	0.1140	5.28
EfficientNetB0	0.5730	0.0930	3.08
EfficientNetB4	0.5670	0.1120	8.16
GhostNet	0.5580	0.0810	1.85
RegNetX-400MF	0.5570	0.1280	1.46
RegNetX-800MF	0.5560	0.0860	2.41
RepVGG	0.5680	0.1650	3.57
ResNeXt	0.5730	0.0850	9.67
ShuffleNetV2	**0.5530**	**0.0970**	**0.80**
SqueezeNet	0.5520	0.0800	1.22
Xception	0.5630	0.0820	10.07

Table 3. Performance comparison of deep learning models based on memory Usage, energy efficiency and inference Time. Significant values are in bold.

accuracy and balanced performance across key predictive metrics, placing them at the highest level despite their relatively higher computational demands. Xception, EfficientNetB4, and SqueezeNet follow closely, offering robust predictive capabilities paired with moderate operational efficiency that reinforces their overall effectiveness. In the next tier, ShuffleNetV2, EfficientNetB0, and RegNetX-800MF provide satisfactory outcomes, with ShuffleNetV2 standing out for its rapid inference speed and efficient resource utilization. In contrast, GhostNet and RegNetX-400MF exhibit more modest predictive performance, which limits their overall utility in applications requiring high accuracy. Given that predictive performance remains the most critical factor, models that combine superior accuracy with reasonable efficiency are preferred for complex macrofungi classification tasks. Collectively, this analysis underscores the trade-offs between computational efficiency and predictive strength, emphasizing that a model's ability to deliver reliable and consistent predictions is paramount in guiding model selection for real-world applications (Tables 2 and 3).

Three different interpretability methods Grad-CAM, thresholded Grad-CAM, and Guided Backpropagation were applied to each of our deep learning models using sample images from three macrofungi species: *Battarrea phalloides*, *Crucibulum laeve*, *and Tulostoma brumale* (Fig. 10). These visualizations revealed the regions most relevant to classification, with Grad-CAM providing coarse localization of discriminative structures, thresholded Grad-CAM isolating the most salient areas, and Guided Backpropagation capturing fine-grained details. Collectively, the methods demonstrated that the networks consistently focused on morphologically distinctive features of each species, thereby enhancing confidence in the models' decision-making processes.

When comparing all models' visualizations encompassing GradCAM, thresholded GradCAM, and Guided Backpropagation several distinct patterns emerge. DenseNet121, ResNeXt, and EfficientNetB4 tend to produce more coherent and concentrated heatmaps, suggesting robust feature extraction that consistently emphasizes crucial macrofungi characteristics. GhostNet, SqueezeNet, and RepVGG show moderate dispersion in their saliency maps, balancing attention across both cap and stipe regions, albeit with occasional gaps or diffuse highlights. ShuffleNetV2 and RegNetX-400MF exhibit sharper, more localized activations, indicating a tendency to focus on a limited set of highly discriminative features, whereas RegNetX-800MF appears to integrate a broader range of visual cues, though sometimes less uniformly. EfficientNetB0 presents slightly broader yet less sharply defined focus areas, while Xception concentrates strongly on the upper portion of the cap, revealing a targeted approach to identifying class-relevant traits. Overall, these differences reflect how each architecture prioritizes and processes morphological details, offering diverse strategies for macrofungi classification (Figs. 11, 12, 13, 14, 15, 16, 17, 18, 19, 20 and 21).

Discussion

This study presents a comprehensive analysis of deep learning models applied to the classification of gasteroid macrofungi species. A total of 11 different CNN architectures were evaluated, with DenseNet121 achieving the highest performance, reaching 96.11% accuracy and an F1-score of 96.09%. ResNeXt followed with 95.00% accuracy, and RepVGG with 93.89%. In contrast, RegNetX-400MF recorded the lowest accuracy at 79.44%. According to ROC-AUC values, DenseNet121 (99.89%) and ResNeXt (99.90%) demonstrated near-perfect class separation. While models like ShuffleNetV2 offered rapid inference times (~0.80 s) suitable for real-time deployment, models such as Xception, despite their high accuracy, presented higher latency (~10.07 s), which may limit their operational use. Memory usage ranged between 55% and 57%, while energy efficiency peaked at 16.5% for RepVGG. These findings highlight the diverse advantages and limitations of different architectures in terms of classification accuracy, computational efficiency, and interpretability.

This approach is not limited to gasteroid fungi but can be extended to other biological structures with similar microscopic or macroscopic characteristics[48]. Examples include lichens, pollen grains, leaf tissues, or bacterial colony morphologies, all of which could be classified effectively using similar architectures[49]. Moreover, datasets derived from light microscopy or scanning electron microscopy (SEM) could also be analyzed with these models[50]. Expanding the study with different datasets and imaging systems would enhance the generalizability of the models and their applicability in biological research[51]. Such applications may also contribute directly to biodiversity monitoring, ecosystem modeling, and conservation biology.

The comparative assessment of the applied algorithms clearly demonstrates how architectural differences affect classification performance. For instance, DenseNet121, with its densely connected layers, facilitates



Fig. 10. Original images of *Battarrea phalloides* (I), *Crucibulum laeve* (II), and *Tulostoma brumale* (III), presented from left to right.

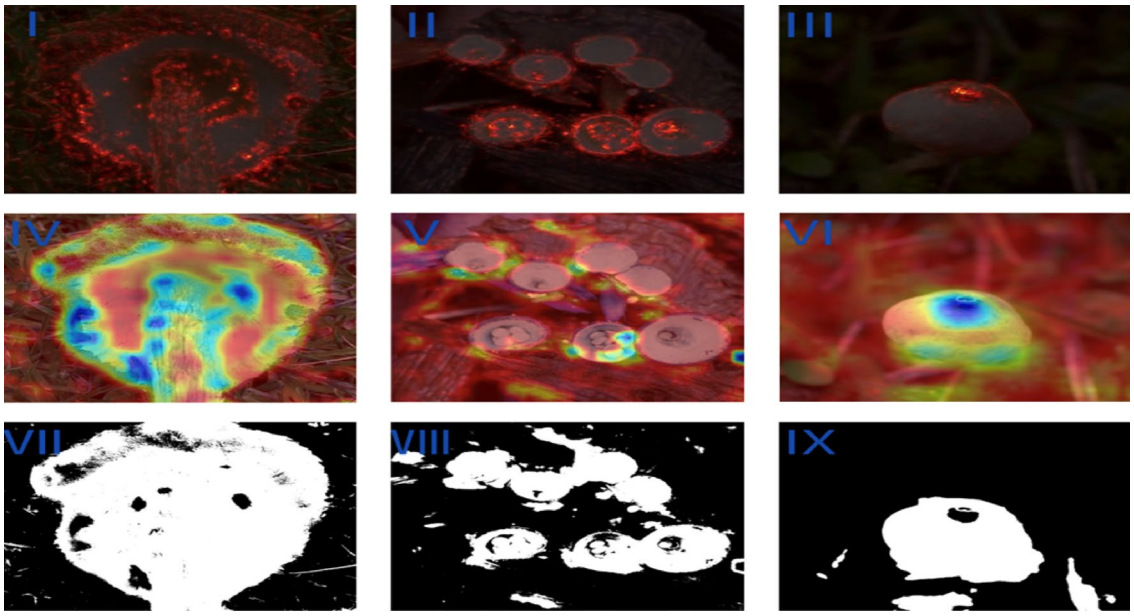

Fig. 11. Visualization of three interpretability methods Guided Backpropagation, GradCAM, and thresholded GradCAM applied to DenseNet121 for three macrofungi species. The top row left to right shows guided backpropagation outputs for *Battarrea phalloides*, *Crucibulum laeve*, and *Tulostoma brumale*; the middle row displays GradCAM overlays highlighting salient regions; and the bottom row presents thresholded GradCAM maps that isolate the most influential areas above an intensity value of 100.

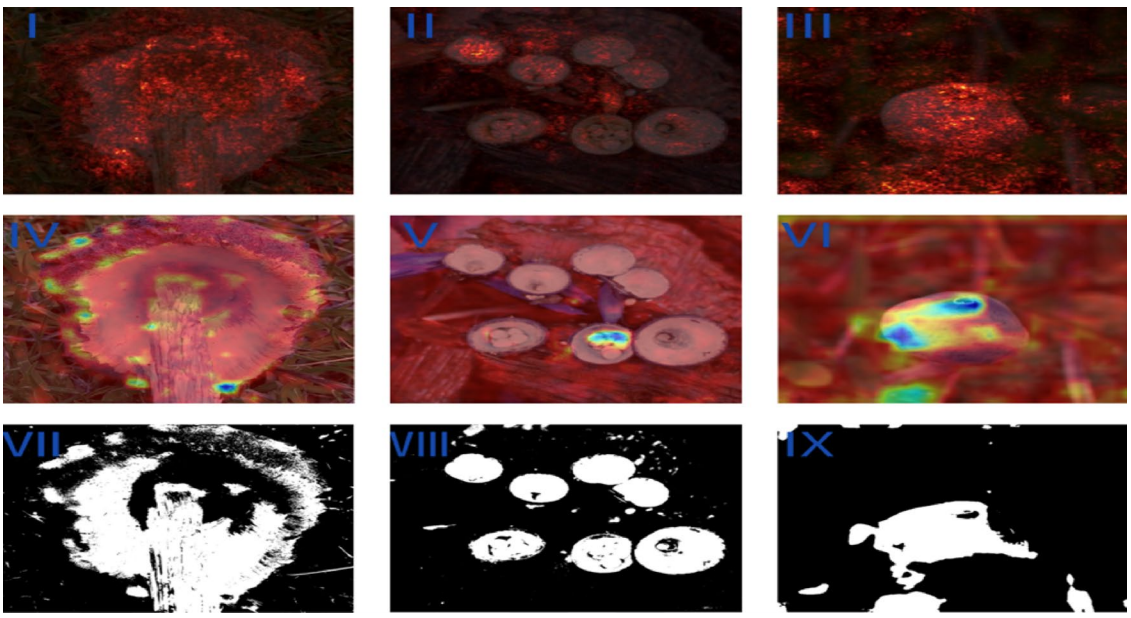

Fig. 12. Visualization of three interpretability methods guided backpropagation, GradCAM, and thresholded GradCAM applied to EfficientNetB0 for three macrofungi species. The top row left to right shows guided backpropagation outputs for *Battarrea phalloides*, *Crucibulum laeve*, and *Tulostoma brumale*; the middle row displays GradCAM overlays highlighting salient regions; and the bottom row presents thresholded GradCAM maps that isolate the most influential areas above an intensity value of 100.

information flow and excels especially in data-limited contexts[52]. The EfficientNet series achieves high accuracy with efficient resource utilization through compound scaling strategies[53]. RepVGG, with its re-parameterizable design, offers balanced performance across training and inference phases[54]. Conversely, lightweight models such as ShuffleNetV2, while offering speed and low memory usage, showed only moderate classification

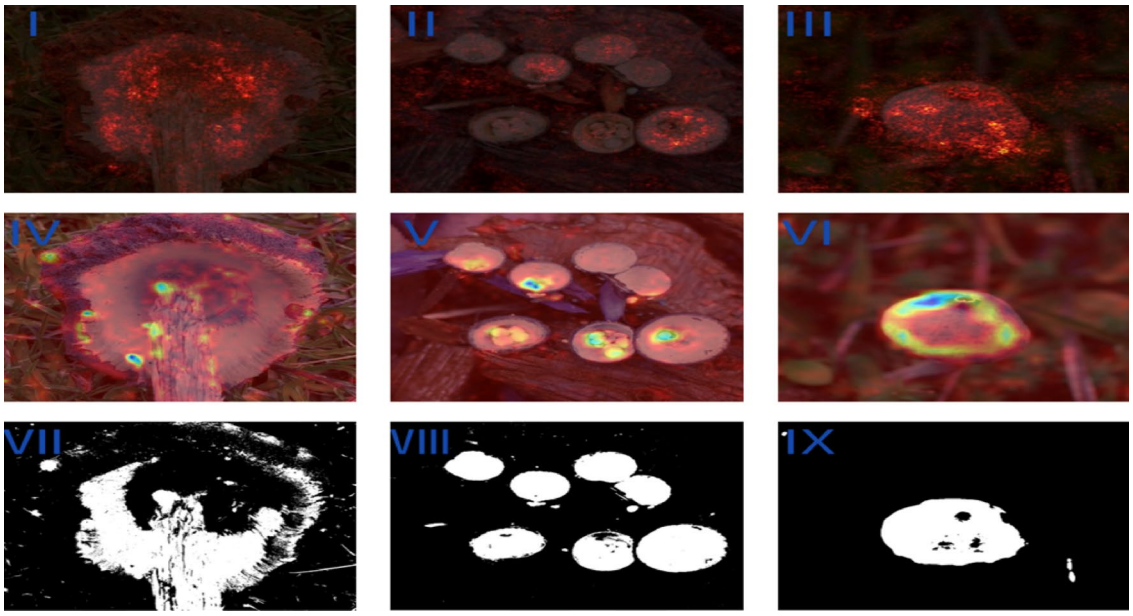

Fig. 13. Visualization of three interpretability methods guided backpropagation, GradCAM, and thresholded GradCAM applied to EfficientNetB4 for three macrofungi species. The top row left to right shows guided backpropagation outputs for *Battarrea phalloides*, *Crucibulum laeve*, and *Tulostoma brumale;* the middle row displays GradCAM overlays highlighting salient regions; and the bottom row presents thresholded GradCAM maps that isolate the most influential areas above an intensity value of 100.

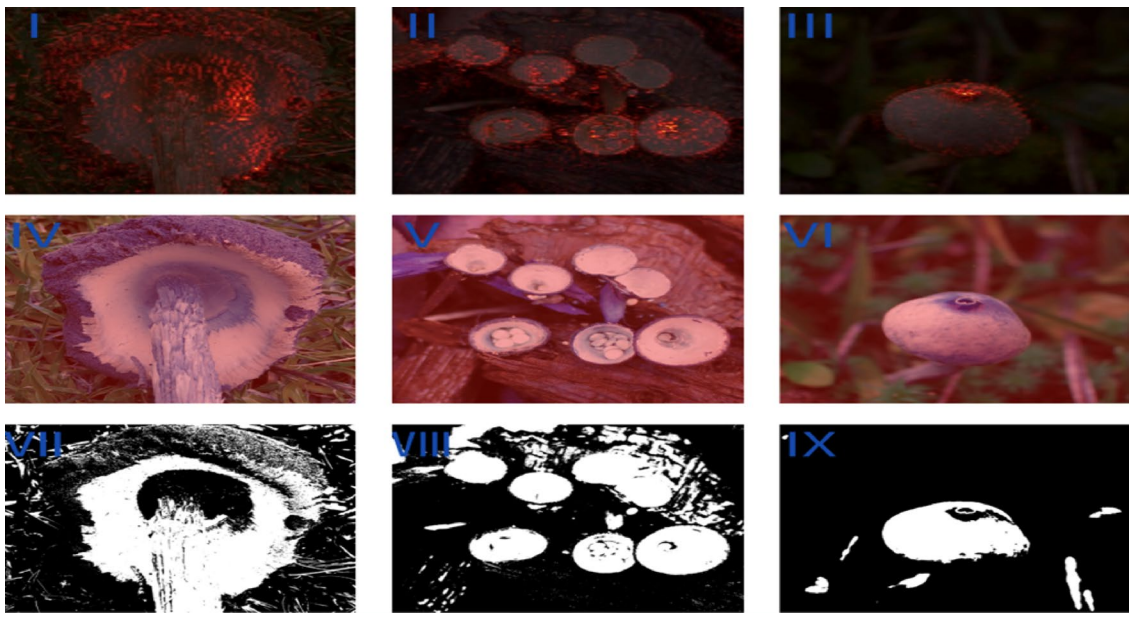

Fig. 14. Visualization of three interpretability methods guided backpropagation, GradCAM, and thresholded GradCAM applied to GhostNet for three macrofungi species. The top row left to right shows guided backpropagation outputs for *Battarrea phalloides*, *Crucibulum laeve*, and *Tulostoma brumale*; the middle row displays GradCAM overlays highlighting salient regions; and the bottom row presents thresholded GradCAM maps that isolate the most influential areas above an intensity value of 100.

performance[55]. These differences emphasize that in model selection, operational factors such as inference time, power consumption, and hardware compatibility should be considered alongside accuracy.

Deep learning-based classification approaches like those used in this study are not confined to biological contexts they can be readily transferred to medicine and engineering[56]. For instance, CNN-based models can be employed in cancer histopathology to automatically classify tumor subtypes[57]. Likewise, SEM images of

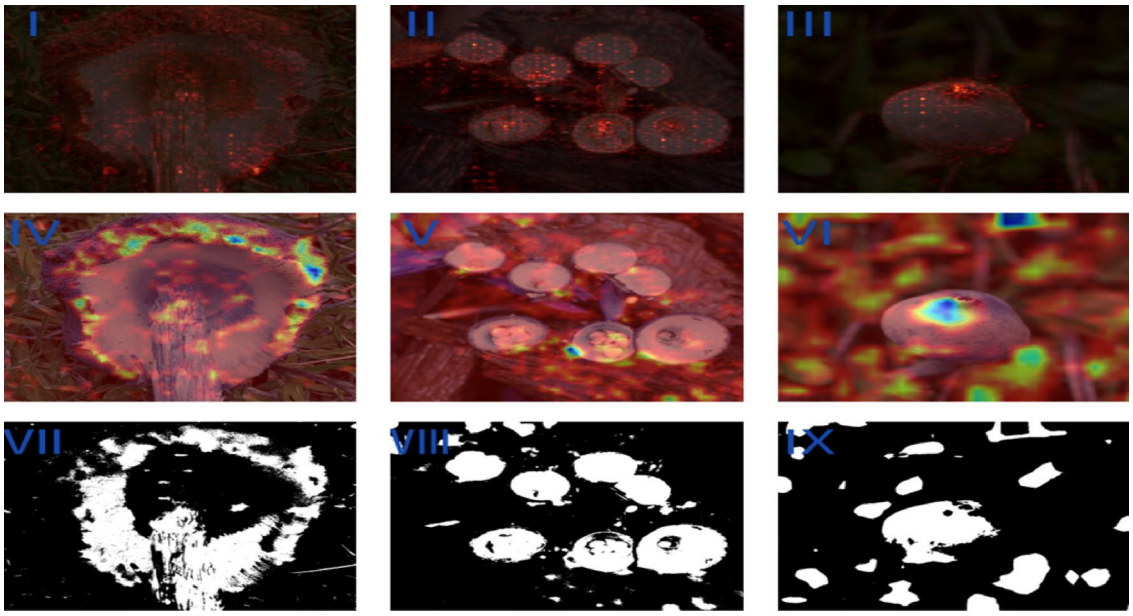

Fig. 15. Visualization of three interpretability methods Guided Backpropagation, GradCAM, and thresholded GradCAM applied to RegNetX-400MF for three macrofungi species. The top row left to right shows Guided Backpropagation outputs for *Battarrea phalloides*, *Crucibulum laeve*, and *Tulostoma brumale*; the middle row displays GradCAM overlays highlighting salient regions; and the bottom row presents thresholded GradCAM maps that isolate the most influential areas above an intensity value of 100.

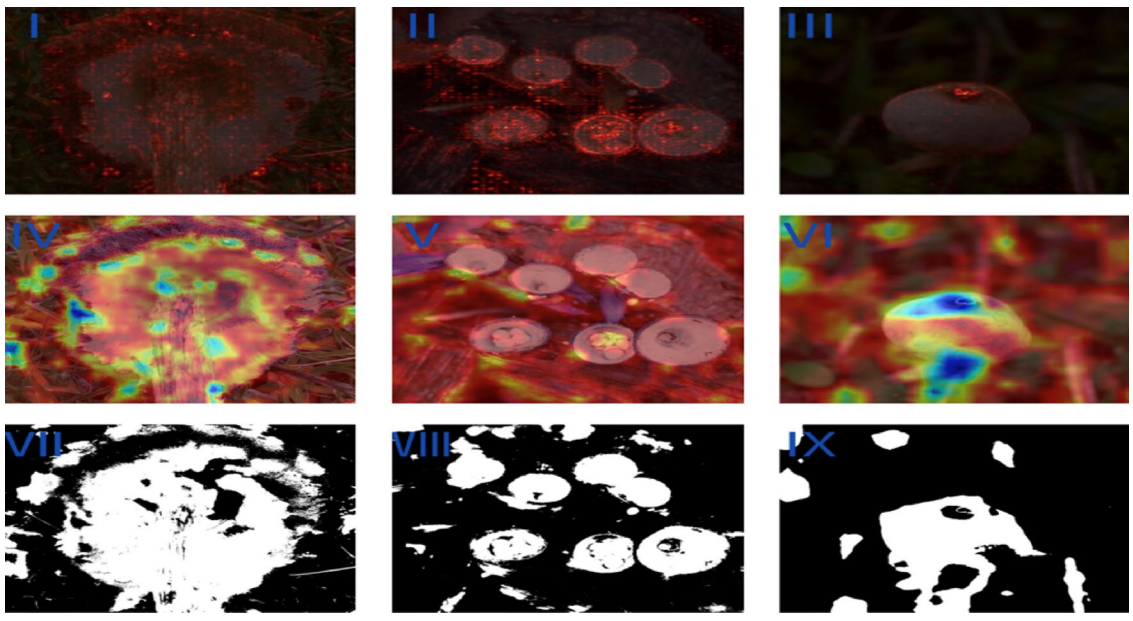

Fig. 16. Visualization of three interpretability methods guided backpropagation, GradCAM, and thresholded GradCAM applied to RegNetX-800MF for three macrofungi species. The top row left to right shows guided backpropagation outputs for *Battarrea phalloides*, *Crucibulum laeve*, and *Tulostoma brumale;* the middle row displays GradCAM overlays highlighting salient regions; and the bottom row presents thresholded GradCAM maps that isolate the most influential areas above an intensity value of 100.

implant materials, polymer surfaces, or nanostructured biomaterials can be effectively analyzed using similar networks. In engineering applications, such models can aid in analyzing fracture surfaces of composite materials or detecting micro-defects in metal surfaces[58]. The comprehensive algorithm comparison and interpretability analyses presented in this study offer a solid foundation for interdisciplinary applications of artificial intelligence, spanning biology, medicine, and materials science.

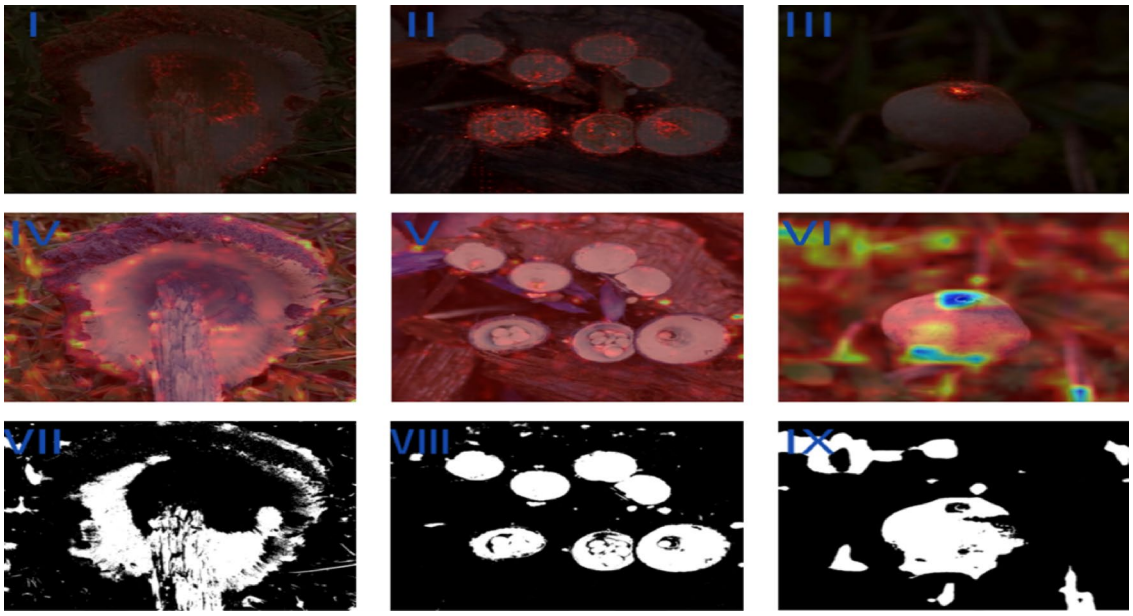

Fig. 17. Visualization of three interpretability methods guided backpropagation, GradCAM, and thresholded GradCAM applied to RepVGG for three macrofungi species. The top row left to right shows guided backpropagation outputs for *Battarrea phalloides*, *Crucibulum laeve*, and *Tulostoma brumale*; the middle row displays GradCAM overlays highlighting salient regions; and the bottom row presents thresholded GradCAM maps that isolate the most influential areas above an intensity value of 100.

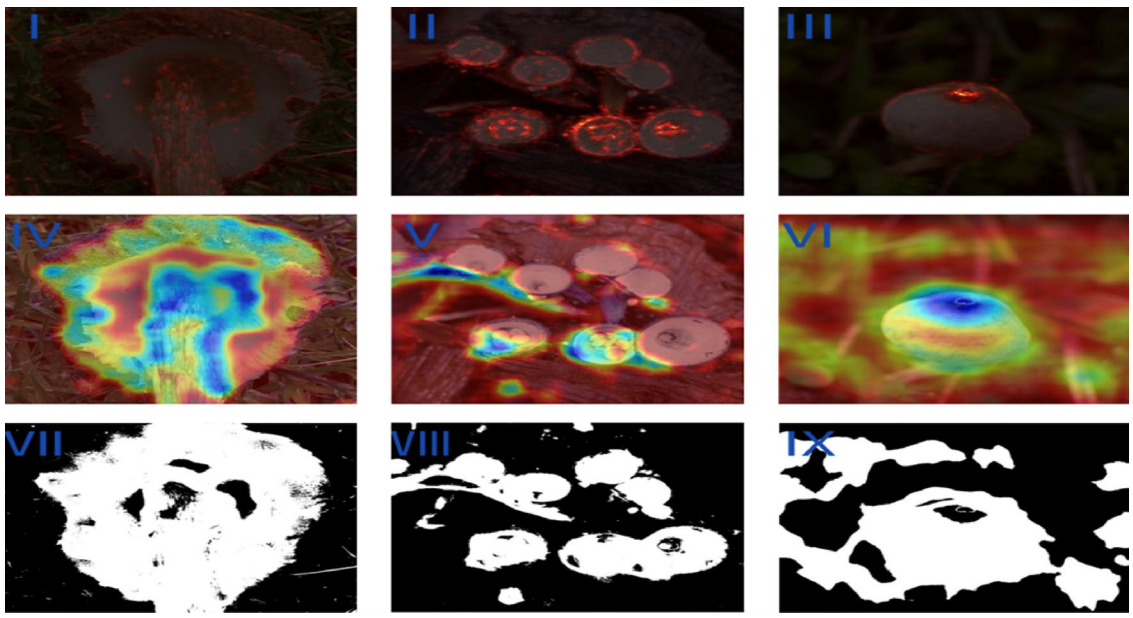

Fig. 18. Visualization of three interpretability methods guided backpropagation, GradCAM, and thresholded GradCAM applied to ResNeXt for three macrofungi species. The top row left to right shows guided backpropagation outputs for *Battarrea phalloides*, *Crucibulum laeve*, and *Tulostoma brumale*; the middle row displays GradCAM overlays highlighting salient regions; and the bottom row presents thresholded GradCAM maps that isolate the most influential areas above an intensity value of 100.

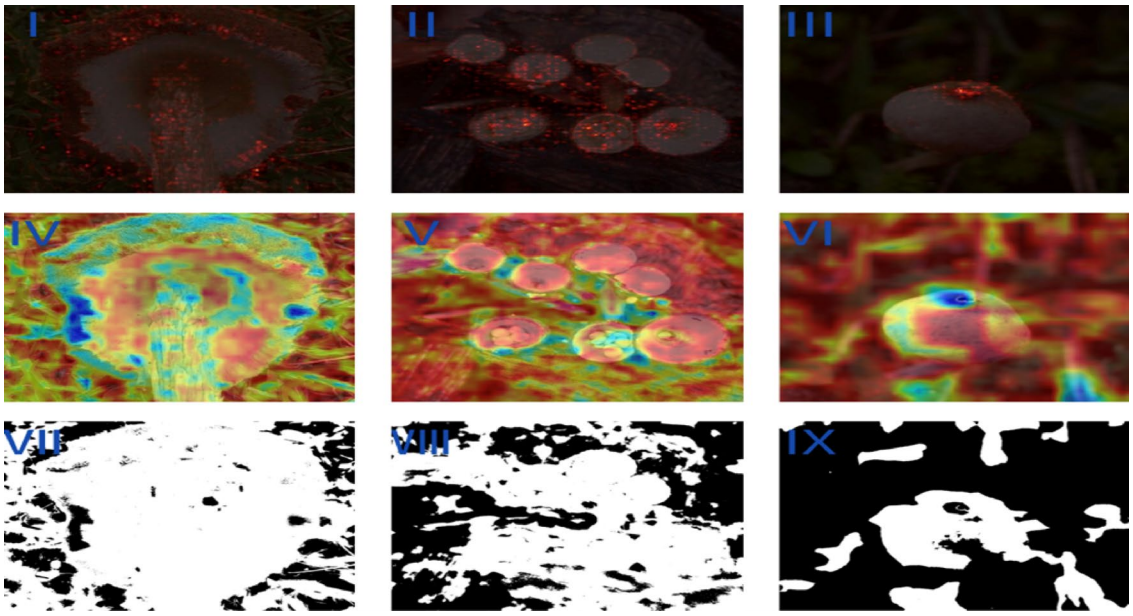

Fig. 19. Visualization of three interpretability methods Guided Backpropagation, GradCAM, and thresholded GradCAM applied to ShuffleNet for three macrofungi species. The top row left to right shows Guided Backpropagation outputs for *Battarrea phalloides*, *Crucibulum laeve*, and *Tulostoma brumale*;; the middle row displays GradCAM overlays highlighting salient regions; and the bottom row presents thresholded GradCAM maps that isolate the most influential areas above an intensity value of 100.

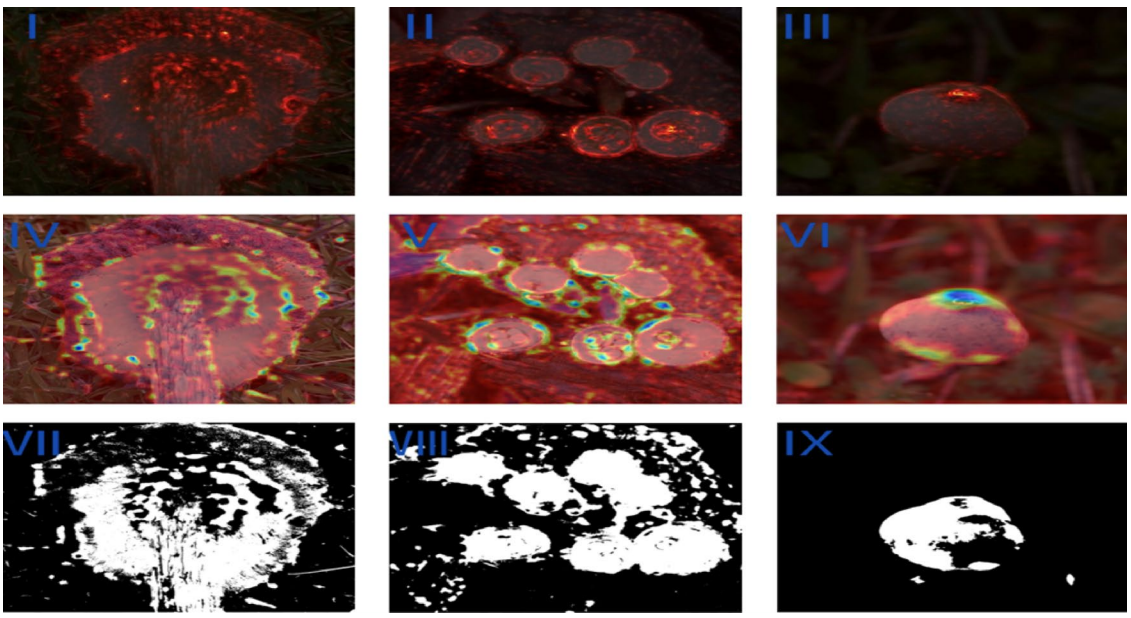

Fig. 20. Visualization of three interpretability methods Guided Backpropagation, GradCAM, and thresholded GradCAM applied to SqueezeNet for three macrofungi species. The top row left to right shows Guided Backpropagation outputs for *Battarrea phalloides*, *Crucibulum laeve*, and *Tulostoma brumale*; the middle row displays GradCAM overlays highlighting salient regions; and the bottom row presents thresholded GradCAM maps that isolate the most influential areas above an intensity value of 100.

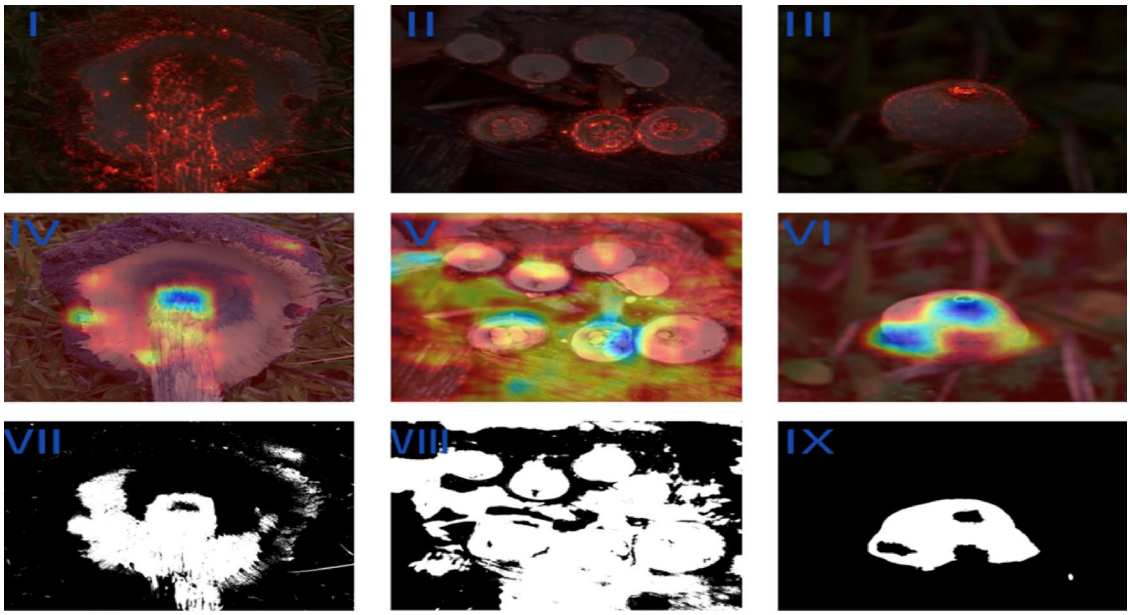

Fig. 21. Visualization of three interpretability methods guided backpropagation, GradCAM, and thresholded GradCAM applied to Xception for three macrofungi species. The top row left to right shows guided backpropagation outputs for *Battarrea phalloides*, *Crucibulum laeve*, and *Tulostoma brumale*; the middle row displays GradCAM overlays highlighting salient regions; and the bottom row presents thresholded GradCAM maps that isolate the most influential areas above an intensity value of 100.

Conclusion

The results of this study demonstrate that deep learning models can effectively classify six gasteroid macrofungi species with high precision and reliability. Among the evaluated architectures, DenseNet121 achieved the highest classification accuracy at 96.11%, with an F1-score of 96.09%, AUC of 99.89%, and a specificity of 99.22%, indicating superior discriminative power and low false positive rates. ResNeXt followed closely with 95.00% accuracy and 99.90% AUC, while RepVGG recorded 93.89% accuracy and the highest energy efficiency at 16.5%, demonstrating a favorable trade-off between performance and operational cost. In contrast, RegNetX-400MF showed the lowest performance with 79.44% accuracy and higher log loss, indicating limitations in learning complex features. Additionally, ShuffleNetV2, despite its moderate accuracy (87.22%), was the fastest model (0.80 s inference time), highlighting its real-time application potential. These findings emphasize that model selection must be aligned with the target use-case, balancing predictive strength, computational efficiency, and interpretability.

Given the model's proven ability to learn from high-resolution macrofungi images, this approach could be extended to a variety of biological classification tasks using microscopic imagery. For instance, the classification of fungal spores, tree and plant pollen, and the identification of rare or endemic botanical species from microscopic or semi-microscopic textures could benefit from similar convolutional neural network architectures. Such applications would be particularly valuable in fields like ecological monitoring, conservation biology, and palynology, where accurate identification plays a crucial role. By adapting these models to new datasets and incorporating domain-specific features, this study paves the way for scalable and explainable AI systems in environmental diagnostics and biological image analysis.

Data availability

Data are available from the corresponding author upon request.

Received: 10 June 2025; Accepted: 9 October 2025
Published online: 17 November 2025

References

1. Calonge, F. D. Gasteromycetes. I. Lycoperdales nidulariales, phallales, sclerodermatales, tulostomales. *Fl Mycol. Iber.* **3**, 1 (1998).
2. Pegler, D. N., Spooner, B. M. & Læssøe, T. British puffballs, earthstars and stinkhorns: An account of the British gasteroid fungi. *(No Title)* (1995).
3. Kraisitudomsook, N., Healy, R. A. & Smith, M. E. Molecular systematics and taxonomic overview of the bird's nest fungi (Nidulariaceae). *Fungal Biol.* **125**(9), 693–703 (2021).
4. Patel, R. R. & Rajput, K. S. Molecular and morphological studies on some gasteroid (Basidiomycota) fungi from Western India. *Asian J. Mycol.* **6**(2), 72–88 (2023).
5. Finy, P. et al. Exploring diversity within the genus Tulostoma (Basidiomycota, Agaricales) in the Pannonian sandy steppe: Four fascinating novel species from Hungary. *MycoKeys* **100**, 153 (2023).

6. Vellinga, E. C., Sysouphanthong, P. & Hyde, K. D. The family agaricaceae: Phylogenies and two new white-spored genera. *Mycologia* **103**(3), 494–509 (2011).
7. Cruz, R., Assis, N., Silva, M. & Baseia, I. Revision of the genus Cyathus (Basidiomycota) from the herbaria of Northeast Brazil. *Mycosphere* **5**, 531–540 (2014).
8. Akata, I., Altuntaş, D., Sahin, E., Alli, H. & Kabaktepe, Ş. A note on Battarrea phalloides in Turkey. *Mantar Dergisi*. **12**, 1–9 (2021).
9. Hobart, C. Tulostoma fimbriatum new to the UK. *Field Mycol.* **13**, 81–83 (2012).
10. Ekinci, F. et al. Classification of Mycena and Marasmius species using deep learning models: An ecological and taxonomic approach. *Sensors* **25**, 1642 (2025).
11. Ozsari, S. et al. Deep learning-based classification of macrofungi: Comparative analysis of advanced models for accurate fungi identification. *Sensors* **24**, 7189 (2024).
12. GBIF.org. GBIF Occurrence Download 02 February (2025). https://doi.org/10.15468/dl.ps5czs
13. Alomar, K., Aysel, H. I. & Cai, X. Data augmentation in classification and segmentation: A survey and new strategies. *J. Imaging*. **9**, 46 (2023).
14. Shen, J., Su, P. C., Cheung, S. S. & Zhao, J. Virtual mirror rendering with stationary rgb-d cameras and stored 3-d background. *IEEE Trans. Image Process.* **22**, 3433–3448 (2013).
15. Oakley, J. P. & Bu, H. Correction of simple contrast loss in color images. *IEEE Trans. Image Process.* **16**, 511–522 (2007).
16. Kanwal, N., Pérez-Bueno, F., Schmidt, A., Engan, K. & Molina, R. The devil is in the details: Whole slide image acquisition and processing for artifacts detection, color variation, and data augmentation: A review. *IEEE Access.* **10**, 58821–58844 (2022).
17. Paszke, A. et al. PyTorch: An imperative style, high-performance deep learning library. In *Proceedings of the 33rd Conference on Neural Information Processing Systems (NeurIPS 2019)* 8024–8035 (2019). https://papers.nips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library
18. TorchVision Contributors. *TorchVision: PyTorch's Computer Vision Library* (2016). Retrieved from https://pytorch.org/vision/
19. Wightman, R. *PyTorch Image Models (timm) [GitHub repository]* (Zenodo, 2019). https://doi.org/10.5281/zenodo.4414861
20. Harris, C. R. et al. Array programming with numpy. *Nature* **585**, 357–362 (2020).
21. Pedregosa, F. Scikit-learn: Machine learning in python. *J. Mach. Learn. Res.* **12**, 2825–2830 (2011).
22. Rodolà, G. *psutil: A Cross-Platform Library for Process and System Monitoring in Python* (2009). Retrieved from https://psutil.readthedocs.io
23. Python Software Foundation. *Python Standard Library Documentation (os, time, json)* (2023). Retrieved from https://docs.python.org/(
24. Zebari, N. A., Alkurdi, A. A., Marqas, R. B. & Salih, M. S. Enhancing brain tumor classification with data augmentation and densenet121. *Acad. J. Nawroz Univ.* **12**, 323–334 (2023).
25. Raza, A. et al. Enhancing brain tumor classification with transfer learning: Leveraging DenseNet121 for accurate and efficient detection. *Int. J. Imaging Syst. Technol.* **34**, e22957 (2024).
26. Priya, B. S. et al. in *2nd International Conference on Artificial Intelligence Trends and Pattern Recognition (ICAITPR)*. 1–6 (IEEE, 2024).
27. Narmadha, D., Majaw, E. A. & Sundar, G. N. In *Synergizing Data Envelopment Analysis and Machine Learning for Performance Optimization in Healthcare* 311–338 (IGI Global Scientific Publishing, 2025).
28. Manole, I., Butacu, A. I., Bejan, R. N. & Tiplica, G. S. Enhancing dermatological diagnostics with efficientnet: A deep learning approach. *Bioengineering* **11**, 810 (2024).
29. JS, N. Brain tumor segmentation using multi-scale attention U-Net with EfficientNetB4 encoder for enhanced MRI analysis. *Sci. Rep.* **15**, 9914 (2025).
30. Chi, J., Guo, S., Zhang, H. & Shan, Y. L-GhostNet: Extract better quality features. *IEEE Access.* **11**, 2361–2374 (2023).
31. Dong, H. et al. Lightweight ghost enhanced feature attention network: An efficient intelligent fault diagnosis method under various working conditions. *Sensors* **24**, 3691 (2024).
32. Mu, L. et al. A fine-grained end-to-end latency optimization framework for wireless collaborative inference. *IEEE Internet Things J.* **11**, 5840–5853 (2023).
33. Hu, L., Yao, C., Huang, J., Liu, J. & Wang, G. Highly Robust Synthetic Aperture Radar Target Recognition Method Based on Simulation Data Training. *Int. J. Antennas Propag.* **2022**, 7537732. (2022).
34. Hu, L., Gao, L., Liu, Z. & Feng, W. Scalable frame resolution for efficient continuous sign Language recognition. *Pattern Recogn.* **145**, 109903 (2024).
35. Ding, X. et al. in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* 13733–13742.
36. Lin, K., Xiao, S. & Shang, K. in *IEEE International Conference on Cognitive Computing and Complex Data (ICCD)*. 77–82 (IEEE, 2024).
37. Li, G., Ren, X., Qiao, W., Ma, B. & Li, Y. Automatic bridge crack identification from concrete surface using resnext with postprocessing. *Struct. Control Health Monit.* **27**, e2620 (2020).
38. Lak, A. J., Boostani, R., Alenizi, F. A., Mohammed, A. S. & Fakhrahmad, S. M. RoBERTa, resnext and BiLSTM with self-attention: the ultimate trio for customer sentiment analysis. *Appl. Soft Comput.* **164**, 112018 (2024).
39. Dutta, M. & Saini, J. in *1st International Conference on Trends in Engineering Systems and Technologies (ICTEST)*. 1–8 (IEEE, 2024).
40. Huang, Y., Lu, Y., Fan, L., Xiang, K. & Ma, H. Fault arc detection method based on improved ShuffleNet V2 network. *Processes* **13**, 135 (2025).
41. Tsivgoulis, M., Papastergiou, T. & Megalooikonomou, V. An improved squeezenet model for the diagnosis of lung cancer in CT scans. *Mach. Learn. Appl.* **10**, 100399 (2022).
42. Mohan, D., Aravinth, J. & Rajendran, S. Reconstruction of compressed hyperspectral image using squeezenet coupled dense attentional net. *Remote Sens.* **15**, 2734 (2023).
43. Wang, B., Huang, L., Huang, T. & Ye, F. Two-stream xception structure based on feature fusion for deepfake detection. *Int. J. Comput. Intell. Syst.* **16**, 134 (2023).
44. Liu, F. et al. Depth-wise separable convolution attention module for garbage image classification. *Sustainability* **14**, 3099 (2022).
45. Aldegheishem, A. et al. Towards sustainable energy efficiency with intelligent electricity theft detection in smart grids emphasising enhanced neural networks. *IEEE Access.* **9**, 25036–25061 (2021).
46. Raghavan, K. & v, K. Attention guided grad-CAM: An improved explainable artificial intelligence model for infrared breast cancer detection. *Multimed. Tools Appl.* **83**, 57551–57578 (2024).
47. Mersha, M., Lam, K., Wood, J., AlShami, A. & Kalita, J. Explainable artificial intelligence: A survey of needs, techniques, applications, and future direction. *Neurocomputing* **599**, 128111 (2024).
48. Pallua, J. D. et al. Application of micro-computed tomography to microstructure studies of the medicinal fungus hericium coralloides. *Mycologia* **107**, 227–238 (2015).
49. Senanayake, I. et al. Morphological approaches in studying fungi: Collection, examination, isolation, sporulation and preservation. *Mycosphere* **11**, 2678–2754 (2020).
50. Dahy, G., Soliman, M. M., Alshater, H., Slowik, A. & Hassanien, A. E. Optimized deep networks for the classification of nanoparticles in scanning electron microscopy imaging. *Comput. Mater. Sci.* **223**, 112135 (2023).
51. Chen, C. et al. Improving the generalizability of convolutional neural network-based segmentation on CMR images. *Front. Cardiovasc. Med.* **7**, 105 (2020).

52. Alkhalefah, S., AlTuraiki, I. & Altwaijry, N. Advancing diabetic foot ulcer care: AI and generative AI approaches for classification, prediction, segmentation, and detection. *Healthcare* **13** (6), 648 (2025).
53. Zhao, Y. et al. A composite scaling network of EfficientNet for improving spatial domain identification performance. *Commun. Biol.* **7**, 1567 (2024).
54. Liu, Y., Fu, X., Zhou, L. & Li, C. Texture-enhanced framework by differential filter-based re-parameterization for super-resolution on PC/mobile. *Neural Process. Lett.* **55**, 12183–12203 (2023).
55. Mittal, P. A comprehensive survey of deep learning-based lightweight object detection models for edge devices. *Artif. Intell. Rev.* **57**, 242 (2024).
56. Ching, T. et al. Opportunities and obstacles for deep learning in biology and medicine. *J. R. Soc. Interface*. **15**, 20170387 (2018).
57. Chaunzwa, T. L. et al. Deep learning classification of lung cancer histology using CT images. *Sci. Rep.* **11**, 1–12 (2021).
58. Li, T. et al. Contrasting effects of sheep and cattle grazing on foliar fungal diseases by changing plant community characteristics. *Funct. Ecol.* **38**, 1172–1184 (2024).

Acknowledgements
This study was prepared using Eda Kumru's master's thesis. We would like to thank TÜBİTAK Scientist Support Programs Directorate (BİDEB).

Author contributions
Conceptualization, E.K., F.E., A.A., M.S.G., M.S., I.A.; methodology, E.K., F.E., A.A., M.S.G., M.S., I.A.; validation, E.K., F.E., A.A., M.S.G., M.S., I.A.; investigation, E.K., F.E., A.A., M.S.G., M.S., I.A.; resources, E.K., F.E., A.A., M.S.G., M.S., I.A.; data curation, E.K., F.E., A.A., M.S.G., M.S., I.A.; writing – original draft preparation, E.K., F.E., A.A., M.S.G., M.S., I.A.; writing-review and editing, E.K., F.E., A.A., M.S.G., M.S., I.A. All authors have read and agreed to the published version of the manuscript.

Declarations

Competing interests
The authors declare no competing interests.

Consent for publication
All authors have approved the fnal article and consented to publication.

Informed consent
This study did not involve human participants.

Additional information
Correspondence and requests for materials should be addressed to I.A.

Reprints and permissions information is available at www.nature.com/reprints.

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Open Access This article is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License, which permits any non-commercial use, sharing, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if you modified the licensed material. You do not have permission under this licence to share adapted material derived from this article or parts of it. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit http://creativecommons.org/licenses/by-nc-nd/4.0/.

© The Author(s) 2025