scientific reports

Check for updates

OPEN

A new method for *Tomicus* classification of forest pests based on improved ResNet50 algorithm

Caiyi Li[1], Quanyuan Xu[1,2]✉, Ying Lu[1,2]✉, Dan Feng[3], Peng Chen[3], Mengxue Pu[4], Junzhu Hu[1] & Mingyang Wang[1]

Tomicus is a globally significant forestry pest, with Yunnan Province in southwestern China experiencing particularly severe infestations. The morphological differences among *Tomicus* species are minimal, making accurate identification challenging. While traditional molecular identification and morphological recognition methods are reliable, they require specialized personnel and equipment and are time-consuming. For individuals with limited expertise, accurate identification becomes particularly difficult. This highlights the challenge of developing a rapid, efficient, and accurate classification model for *Tomicus*. This study investigates four major *Tomicus* species in Yunnan Province: *Tomicus yunnanensis*, *Tomicus minor*, *Tomicus brevipilosus*, and *Tomicus armandii*. We collected samples from infested pine trees and constructed a dataset comprising 6,371 high-resolution images captured using a handheld microscope. A novel *Tomicus* classification model, DEMNet, was proposed based on an improved ResNet50 architecture. Experimental results demonstrate that DEMNet outperforms ResNet50 across key metrics, achieving a classification accuracy of 92.8%, a parameter count of 1.6 M, and an inference speed of 0.1193 s per image. Specifically, DEMNet reduces the parameter count by 90% while improving classification accuracy by 9.5%. Its lightweight and high-precision design makes DEMNet highly suitable for deployment on embedded devices, offering significant potential for real-time *Tomicus* identification and pest management applications.

Tomicus (Latreille) (Coleoptera: Curculionidae: Scolytinae) is one of the most significant forestry pest species globally, widely distributed across much of the Eurasian continent, where it primarily infests coniferous trees, especially those in the Pinaceae family[1]. The spread of *Tomicus* worldwide has led to extensive forest die-offs, intensifying issues related to forest health, in southwestern China, *Tomicus* is especially prevalent in pine forests, triggering regional outbreaks of forestry pests[2]. In recent years, the scale of infestations has expanded due to climatic changes and insufficient forest management practices, adversely affecting forest growth, Yunnan Province has been particularly impacted by *Tomicus* infestations[3]. The province's abundant pine resources have contributed to the pest's high reproductive capacity and adaptability in the region[4]. Although Yunnan boasts rich forestry resources, the widespread and recurrent outbreaks of *Tomicus* pose a significant threat to its forest ecosystems[5]. This study focuses on four *Tomicus* species that severely damage pine trees in Yunnan: *Tomicus yunnanensis* Kirkendall & Faccoli[6], *Tomicus minor* (Hartig)[7], *Tomicus brevipilosus* (Eggers)[8], and *Tomicus armandii* Li & Zhang[9]. The elytral puncture patterns and setae length among these species display only slight differences, complicating their identification. Traditional *Tomicus* identification methods primarily involve molecular analysis and morphological observation. For example, Wu et al.[10] differentiated *T.yunnanensis* and *T.minor* using morphological traits and molecular data, while also examining pheromone-mediated regulatory mechanisms influencing intra and interspecific behaviors during feeding on young shoots. Li et al.[9] identified the newly classified *T.armandii* through genetic distance measurement using the D2 fragment of 28SrDNA and phylogenetic analysis, as well as through two morphological traits. Similarly. Kirkendall et al.[6] classified species within the *Tomicus* genus using morphological features observed under transmitted light alongside molecular sequences. While these methods are precise and reliable, they require expertise and equipment and are often

[1]College of Big Data and Intelligent Engineering, Southwest Forestry University, Kunming 650224, China. [2]Key Laboratory of Forestry and Ecological, Big Data State Forestry Administration on Southwest Forestry University, Kunming 650024, China. [3]Yunnan Academy of Forestry and Grassland, Kunming, China. [4]College of Ecology and Environmental Sciences, Southwest Forestry University, Kunming 650224, China. ✉email: xuqy@swfu.edu.cn; 80439808@qq.com

time-intensive. For those with limited experience, accurately identifying *Tomicus* species can be challenging in both field and laboratory settings. Therefore, a rapid, efficient, and highly accurate classification method is needed to overcome these identification difficulties.

Species-level identification of *Tomicus* species is of significant ecological and economic importance. First, *Tomicus* is a genus of major forest pests, particularly pathogenic to pine trees, causing severe damage to forest ecosystems. Different *Tomicus* species exhibit distinct biological characteristics and dispersal patterns, and precise species identification is essential for understanding their population dynamics, distribution patterns, and impact on forest health. Second, species identification provides a scientific foundation for developing targeted pest management strategies. Different species may respond differently to various pesticides or control measures, and accurate species identification enables the optimization of pest control efforts, reducing environmental impact and economic losses. Additionally, with the influence of climate change, the distribution range of certain *Tomicus* species may shift, making species-level identification crucial for monitoring and predicting such changes. Therefore, precise identification of *Tomicus* species is a fundamental and necessary step for forest protection, pest management, and ecological monitoring.

With the rapid advancement of computer technology, deep learning has found increasingly widespread applications in fields such as the classification and identification of forestry and agricultural pests[11]. Peng et al.[12] introduced modifications based on MobileNetV2 to address the common challenges in pest classification research, namely the high model complexity and low classification accuracy. He et al.[13] was the first to propose the ResNet architecture, which successfully addressed the vanishing gradient problem in deep networks through the introduction of residual units. Li et al.[14] employed a transfer learning-based ResNet50 in the classification of rice pests, achieving a classification accuracy of 83.2%. Depthwise Separable (DS) convolution is an efficient convolutional operation that is divided into depthwise convolution and pointwise convolution[15], and it has been widely adopted in numerous convolutional neural network architectures. Howard et al.[16] were the first to introduce depthwise separable convolution in the MobileNet model, significantly reducing both the number of parameters and computational cost, although the accuracy on the ImageNet dataset was only 71.7%. Kamal et al.[17] employed this convolutional architecture in a plant disease detection model for leaf images, significantly accelerating the model's convergence speed while reducing the number of parameters by a factor of 29 compared to the VGG model. The attention mechanism is also one of the commonly used techniques in deep learning. Wang et al.[18] proposed an Efficient Channel Attention (ECA) mechanism and integrated it into ResNet50, achieving an improvement of over 2.0% in top-1 accuracy on the ImageNet dataset. Ni et al.[19] incorporated ECA into the RepVGG model to classify six categories: five rice pests and diseases, and healthy rice, resulting in a 1.1% increase in classification accuracy. The ECA mechanism enhances the representation capability of channel information by adaptively selecting convolutional kernel sizes for cross-channel information fusion[20]. MobileNetV3 integrates depthwise separable convolutions with Neural Architecture Search (NAS) techniques, optimizing the design of activation functions and fully connected layers, and is widely used in edge computing and real-time detection scenarios[21]. In deep learning research, achieving high classification accuracy, rapid inference speed, and lightweight network models has consistently been a primary goal for many researchers[22–24]. For instance, Zheng et al.[25] designed the PCNet model based on EfficientNetV2, enabling precise and efficient operation on mobile devices. He et al.[26] constructed a lightweight network called LiteNet, which requires less inter-communication between processing units during distributed training, making it suitable for deployment on resource-constrained mobile devices. Although deep learning is widely applied across various fields, we have not yet identified its use in existing *Tomicus* classification methods. This study fills a technological gap in this area.

Building on previous research, accurately classifying *Tomicus* species using deep learning remains a significant challenge due to the high degree of similarity among their features. Consequently, the ResNet series has become a primary approach for insect image classification. While ResNet meets the basic classification requirements, its capacity to capture fine-grained local features is limited, and it faces ongoing challenges in terms of parameter efficiency and lightweight model design. In response to these challenges, this study improves the ResNet50 model using a high-resolution image dataset of the *Tomicus*, aiming for a lightweight structure and improved classification accuracy. Accordingly, we propose a new classification network model named DEMNet, specifically tailored for *Tomicus* classification. The main contributions and innovations of this study are as follows:

1. By integrating DS convolution instead of standard convolutions and adjusting the output channels for each residual module, this study achieves a lightweight model design that significantly reduces computational complexity.
2. The modified residual structure incorporates the ECA mechanism, enhancing the model's ability to capture fine-grained features and markedly improving classification accuracy. Additionally, the original fully connected layer has been replaced with a MobileNetV3 classifier, which further boosts both classification precision and generalization capability.
3. This study applies the Parametric Rectified Linear Unit (PReLU) activation function globally in place of the original model's Rectified Linear Unit (ReLU), providing the model with greater adaptability in data processing. This adjustment strengthens the model's learning capacity and accelerates convergence, thereby effectively increasing classification accuracy.

The lightweight network model we propose delivers reliable technical support for deployment on embedded devices. This model not only meets the research objectives of rapid inference and high classification accuracy but also provides an effective alternative to traditional methods of molecular identification and morphological recognition, which are often excessively time-consuming. Furthermore, the enhanced accuracy of the model, particularly when utilized by individuals with limited classification experience, underscores its practicality

in real-world applications, especially in fieldwork or resource-constrained environments where it holds considerable value.

Materials and methods
Data acquisition and preprocessing
Data source and acquisition

This study focuses on four species of *Tomicus*, forestry pests that infest the treetops and trunks of various pine trees. Due to the complexity of field environments and the subtlety of target features, we collected infested pine samples from locations in Yuxi, Qujing, Dali, and other areas within Yunnan Province, China. Under laboratory conditions, *Tomicus* specimens were briefly anesthetized by immersion in anhydrous ethanol and subsequently placed under shadowless lighting for imaging. Given their body lengths of no more than 5 mm, we utilized a 100-megapixel handheld IPS high-definition microscope to capture high-resolution images, thereby constructing a high-quality color dataset. Sample images of the four *Tomicus* species are presented in Fig. 1, and their features are detailed in Table 1, with all images in the table magnified by the handheld high-definition microscope.

Data preprocessing and allocation

To enhance the quality of the dataset and address the issue of class imbalance, this study implemented preprocessing techniques. The preprocessing primarily involved data augmentation and balancing the number of samples across categories to improve the model's generalization performance and ensure the scientific rigor of model training, validation, and testing. In terms of data augmentation, we initially enhanced the dataset using a combination of methods such as random cropping, horizontal flipping, rotation, and denoising. In the process of random cropping, the size range of the cropping area is determined in accordance with the distribution of the key regions of the *Tomicus* within the selected images. A cropping window is then randomly selected within this predefined range. This approach enables the model to learn diverse local features of the target. For horizontal flipping, each image is subjected to this transformation with a 50% probability. This is done to assist the model in adapting to variations in the orientation of the target. When it comes to rotation, angles such as -45° and 90° are set. This allows the model to become acquainted with the appearance of the *Tomicus* from different angular perspectives. Finally, the non-local means denoising algorithm is utilized. This algorithm effectively removes the noise introduced during image acquisition, optimizes the image quality, provides higher-quality data for model training, and comprehensively enhances the richness and effectiveness of the dataset. To tackle the class imbalance issue, we selected images from the original data at a specified ratio for augmentation based on different categories. For example, for *T.yunnanensis*, *T.minor*, and *T.brevipilosus*, we randomly selected 30% of the images from each category for data augmentation. In contrast, for *T.armandii*, which had a relatively smaller amount of original data, we selected 50% of the images for augmentation. After data augmentation and balancing, the number of samples in each category is presented in Table 2. The total size of the dataset increased from the original 4,934 images to 6,371 images, while ensuring that the number of images in each category remained at approximately 1,590. Finally, the dataset was divided into training, validation, and test sets in a ratio of 8:1:1, ensuring balance across categories in each set. This approach provided the model with ample learning and evaluation samples, thereby enhancing its reliability and accuracy.

Building the model
ResNet50

He et al.[13] introduced the pivotal concept of residual networks in their research. Unlike traditional neural networks, which primarily focus on learning the direct mapping from input x to output y. residual networks compute this residual $f(x)$ and add it to the original input x (i.e., $f(x) + x$)[27] to produce the final output. This design employs the Skip Connection mechanism, allowing the original input x to participate directly in the output computation. This significantly reduces information loss during propagation through deep networks, thereby decreasing the learning difficulty and complexity of the model. ResNet50 offers advantages over other ResNet architectures in terms of performance, parameter efficiency, and training stability, making it particularly suitable for tasks that require high accuracy and rapid training speed. Consequently, ResNet50 has emerged as the ideal model for the classification of *Tomicus*. This method is not only innovative in theory but has also been empirically validated to substantially enhance the training efficiency and performance of deep neural networks, particularly when handling large-scale datasets and complex tasks.
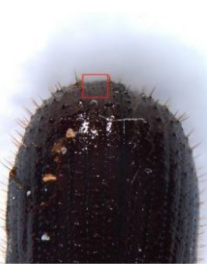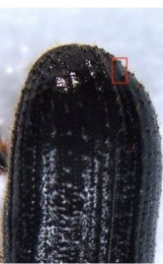
Class	*Tomicus yunnanensis*	*Tomicus minor*	*Tomicus brevipilosus*	*Tomicus armandii*
image samples				

Fig. 1. Four types of *Tomicus* magnified sample images by 100x.

Category	Image	Characteristics	Main affected pine species
T. yunmanensis		The second interval of the declivity on the elytra exhibits two rows of punctures; the tubercular granules on the declivity are covered with long hairs.	*Pinus yunmanensis*
T.minor		The second interval of the declivity on the elytra displays a single row of punctures.	*Pinus massoniana, Pinus yunmanensis, Pinus tabulaeformis Pinus tabulaeformis*
T. brevipilosus		The morphological characteristics are similar to those of *T. yunmanensis*, but the tubercular granules on the declivity of the elytra have short hairs.	*Pinus yunmanensis, Korean pine*
T. armandii		The tubercular granules on the declivity of the elytra are small and widely spaced.	*Pinus armandi*

Table 1. Introduce the characteristics of different types of *Tomicus*, adapted from Kirkendall et al.[6], Li et al.[9] and Wu et al.[10] (The magnification of the image is 100x.).

Class	Class_Name	Before data preprocessing	After data preprocessing
1	*T.yunnanensis*	1232	1598
2	*T.minor*	1225	1593
3	*T.brevipilosus*	1217	1589
4	*T.armandii*	1060	1591

Table 2. Number of pictures of *Tomicus*.
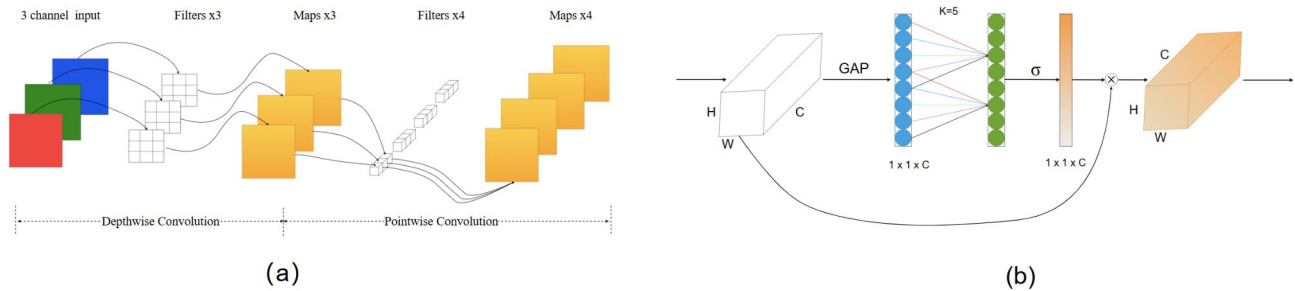
Fig. 2. (**a**) Structure of the depthwise separable convolution. (**b**) Structure of the efficient channel attention module.

Improved residual block

In this study, to achieve a lightweight model, we integrated DS convolution into the residual modules of the *Tomicus* classification model, replacing the standard convolution employed in the original model. DS convolution is composed of two parts: depthwise convolution and pointwise convolution, as shown in Fig. 2(a). For a three-channel color image input, assume the input image is $I \in R^{H \times W \times 3}$, where H is the image height, W is the image width, and 3 represents the number of channels. Depthwise convolution applies independent convolutional kernels to each channel separately. Specifically, let the depthwise convolutional kernel be $K_d \in R^{k \times k \times 1}$ (k is the size of the convolutional kernel). For each channel $i = 1,2,3$, the calculation process of depthwise convolution is as shown in Eq. (1).

$$F_d^i (m, n) = \sum_{u=0}^{k-1} \sum_{v=0}^{k-1} I (m + u, n + v, i) \times K_d (u, v, 1) \tag{1}$$

Where F_d^i represents the feature map of the i channel after depthwise convolution, and (m, n) are the coordinates on the feature map. After a depthwise convolution, feature maps $F_d \in R^{H' \times W' \times 3}$ with the same number of channels as the input are generated (H' and W' are the height and width of the feature maps after convolution). Subsequently, pointwise convolution comes into play, which uses 1×1 convolutional kernels to process the output of depthwise convolution. Let the pointwise convolutional kernel be $K_p \in R^{1 \times 1 \times C}$ (C is the number of pointwise convolutional kernels). The calculation process of pointwise convolution is as shown in Eq. (2).

$$F_p (m, n) = \sum_{c=0}^{C-1} \sum_{i=0}^{3} F_d (m, n, i) \times K_p (0, 0, C) \tag{2}$$

Where F_p represents the feature map after pointwise convolution. As can be seen in the figure, the three feature maps after depthwise convolution are processed by multiple 1×1 convolutional kernels (the case of four 1×1 convolutional kernels is shown in Fig. 2(a). The core objective of pointwise convolution is to integrate information across channels. It combines these feature maps into new feature maps through operations such as weighted summation of feature maps from different channels, effectively achieving information fusion. This design not only significantly reduces the computational load but also maintains the feature-extraction ability of the model to a certain extent, contributing to the realization of the model lightweight goal.

The attention mechanism plays a crucial role in deep learning, enabling the efficient and precise extraction of useful information from large volumes of data[28]. In this study, the ECA was introduced into the newly constructed residual blocks. By adaptively adjusting the weights of channel features, the network can focus on key features, enhancing feature discrimination while effectively suppressing interference from less relevant features. The ECA module enhances the model's ability to capture important features, thereby facilitating the precise identification of *Tomicus* information. Without significantly increasing the computational burden, the ECA module improved the model's classification performance, thereby effectively enhancing classification accuracy. The network architecture is illustrated in Fig. 2 (b).

The principle of the ECA module is as follows: First, global average pooling is applied to the input feature map of size $H \times W \times C$. This operation compresses the spatial information of the feature map into a channel descriptor vector. The dimension of this vector is $1 \times 1 \times C$, representing the global features of each channel.

Next, a one-dimensional convolution of size k is applied, followed by a sigmoid activation function to generate new weights w, facilitating information interaction among channels, as shown in Eq. (3).

$$w = \sigma \left(Conv1D \left(F, k \right) \right) \tag{3}$$

Here, $Conv1D(F, k)$ denotes the one-dimensional convolution operation applied to the input feature F with a kernel size of k, and σ represents the sigmoid activation function. It is widely recognized that the number of channels C is typically set as a power of two and is proportional to the kernel size k in the one-dimensional convolution, as indicated in Eq. (4).

$$C = 2^{(\gamma * k - b)} \tag{4}$$

Therefore, given the number of channels C, the final kernel size k can be obtained, as shown in Eq. (5).

$$k = \left| \frac{\log_2(c)}{\gamma} + \frac{b}{\gamma} \right|_{odd} \tag{5}$$

Where $|t|_{odd}$ denotes the nearest odd number to t, γ is 2, b is 1.

ResNet50 (where "50" denotes the total number of layers in the network architecture (including various types of layers, such as convolutional and pooling layers)) is a deep neural network architecture. The basic structure of its residual block is shown in Fig. 3 (a), where a skip connection directly adds the input to the output. This structure allows the network to learn the residuals between the input and output, simplifying the optimization of the learning objective. Specifically, in the case of layer2, the input data first passes through a convolutional layer with a stride of 1, a 1×1 kernel, and 128 output channels, followed by batch normalization (BN) and ReLU
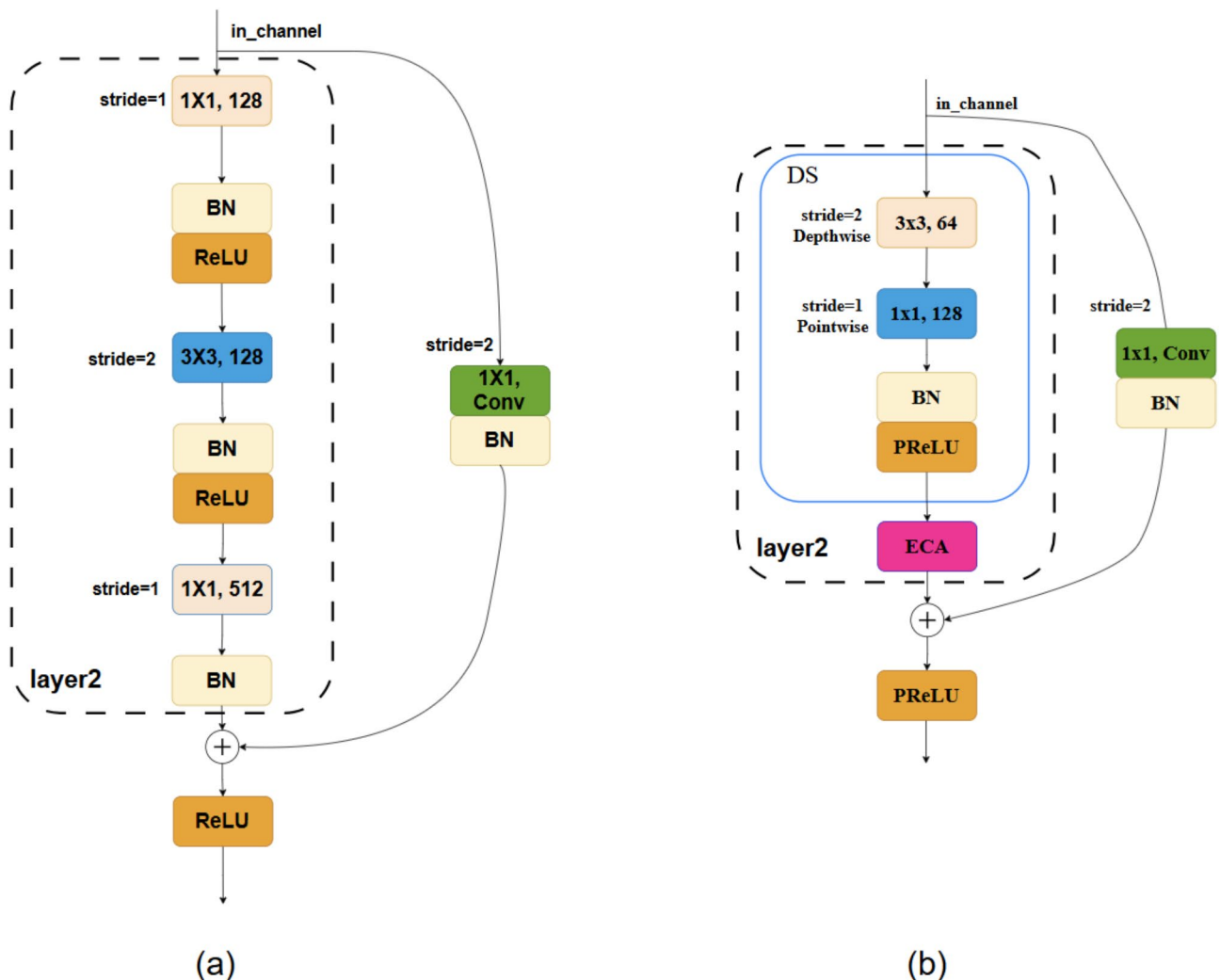
Fig. 3. The residual structure of ResNet50, taking layer2 as an example: (**a**) ResNet50 residual block. (**b**) New residual block.

layers. Next, it passes through a convolutional layer with a stride of 2, a 3×3 kernel, and 128 output channels, followed by BN and ReLU layers. Then, it goes through a convolutional layer with a stride of 1, a 1×1 kernel, and 512 output channels, followed by a BN layer. Finally, a skip connection adds the input to the output, and the result is processed by a ReLU function. This structure enables the network to learn residuals, thereby simplifying the optimization process of the learning objective.

In this study, we improved the residual blocks of ResNet50, and the specific network structure is shown in Fig. 3(b). The improved structure first applies depthwise separable convolutions to the input feature map, including a depthwise convolution with a stride of 2, a 3×3 kernel, and 64 output channels, followed by a pointwise convolution with a stride of 1, a 1×1 kernel, and 128 output channels. The output then passes through BN and PReLU layers, and the resulting feature information is fed into the ECA attention mechanism module to enhance feature selection. The improved structure still utilizes skip connections, where the input passes through a convolutional layer with a stride of 2, a 1×1 kernel, and BN before being added to the output, which is then optimized by PReLU for improved learning.

Modification of the fully connected layer
Compared to the fully connected layer of ResNet50, the MobileNetV3 classifier offers substantial advantages for enhancing classification accuracy due to its lightweight design, effective regularization techniques, adaptive activation functions, and flexible training strategies. Therefore, this study replaces the fully connected layer of ResNet50 with the MobileNetV3 classifier. The classifier effectively mitigates overfitting through components such as fully connected layers, PReLU activation functions, and dropout layers, while using the SoftMax function for probability distribution calculations. This approach maintains model performance while enhancing inference speed, and further reduces the risk of overfitting. This improved method contributes to better classification accuracy for *Tomicus*.

Global application of prelu activation function
An activation function is a nonlinear function that strengthens the nonlinear relationship between the outputs of upper-layer nodes and the inputs of lower-layer nodes in multilayer neural networks[29]. Selecting an appropriate activation function is essential for a specific training model, as it can significantly improve neural network performance[30].

ReLU is the most used activation function, known for effectively mitigating the vanishing gradient problem in deep neural networks. Its introduction has driven substantial advancements in the field of deep learning[31]. The expression is defined as shown in Eq. (6).

$$ReLU\left(x\right) = \left\{ \begin{array}{ll} 0, & x \leq 0 \\ x, & x > 0 \end{array} \right. \tag{6}$$

Here, x is the input. When the input is less than or equal to 0, the gradient of ReLU is 0, indicating that the neuron is 'dead' and cannot update its weights, resulting in information loss. To mitigate this issue, an improved ReLU function known as the PReLU activation function has been proposed. The expression for PReLU is defined in Eq. (7).

$$PReLU\left(x\right) = \left\{ \begin{array}{ll} 0, & x \leq 0 \\ \alpha\, x, & x > 0 \end{array} \right. \tag{7}$$

Here, x is the input, and α is a learnable parameter. The PReLU function represents an enhancement over ReLU, allowing for adaptive learning of parameters from the data, and providing benefits such as accelerated convergence and reduced error rates. To maximize the model's expressiveness, enhance its predictive capability for *Tomicus*, and improve classification accuracy, this study globally replaces the original ReLU function with the PReLU activation function.

Proposed model
To address the design challenges associated with the small size of *Tomicus*, difficulties in enhancing classification accuracy, and the large number of model parameters, this study proposes a lightweight classification model for *Tomicus* based on an improved ResNet50 framework, termed DEMNet. with its structure illustrated in Fig. 4.

As shown in Fig. 4, the overall model retains the residual network framework of ResNet50. Our proposed DEMNet model introduces three significant improvements over ResNet50: First, the original residual block is replaced with the Improved Residual Block shown in (A) of Fig. 4. This module incorporates DS convolution and the ECA attention mechanism. The DS convolution first applies depthwise convolutions to process each input channel separately, extracting channel-specific features, and then performs pointwise convolutions to integrate information across channels. This reduces computational load while effectively facilitating feature interaction. The ECA attention mechanism further processes the feature information, adaptively learning the importance of each channel, thereby enhancing feature selection capability, improving classification accuracy, and reducing computational costs. Secondly, the original fully connected layer of ResNet50 is replaced with the MobileNetV3 classifier shown in (B) of Fig. 4. The MobileNetV3 classifier, which adopts an advanced architectural design, outperforms traditional fully connected layers in terms of feature extraction and processing, achieving superior classification accuracy while reducing the number of parameters and computational cost. Finally, the ReLU activation function in the model is universally replaced with PReLU(C). PReLU automatically adjusts the slope of the negative half-axis based on the data, enhancing the model's adaptability to different data distributions and improving overall predictive capability, further boosting classification accuracy. Additionally, by carefully
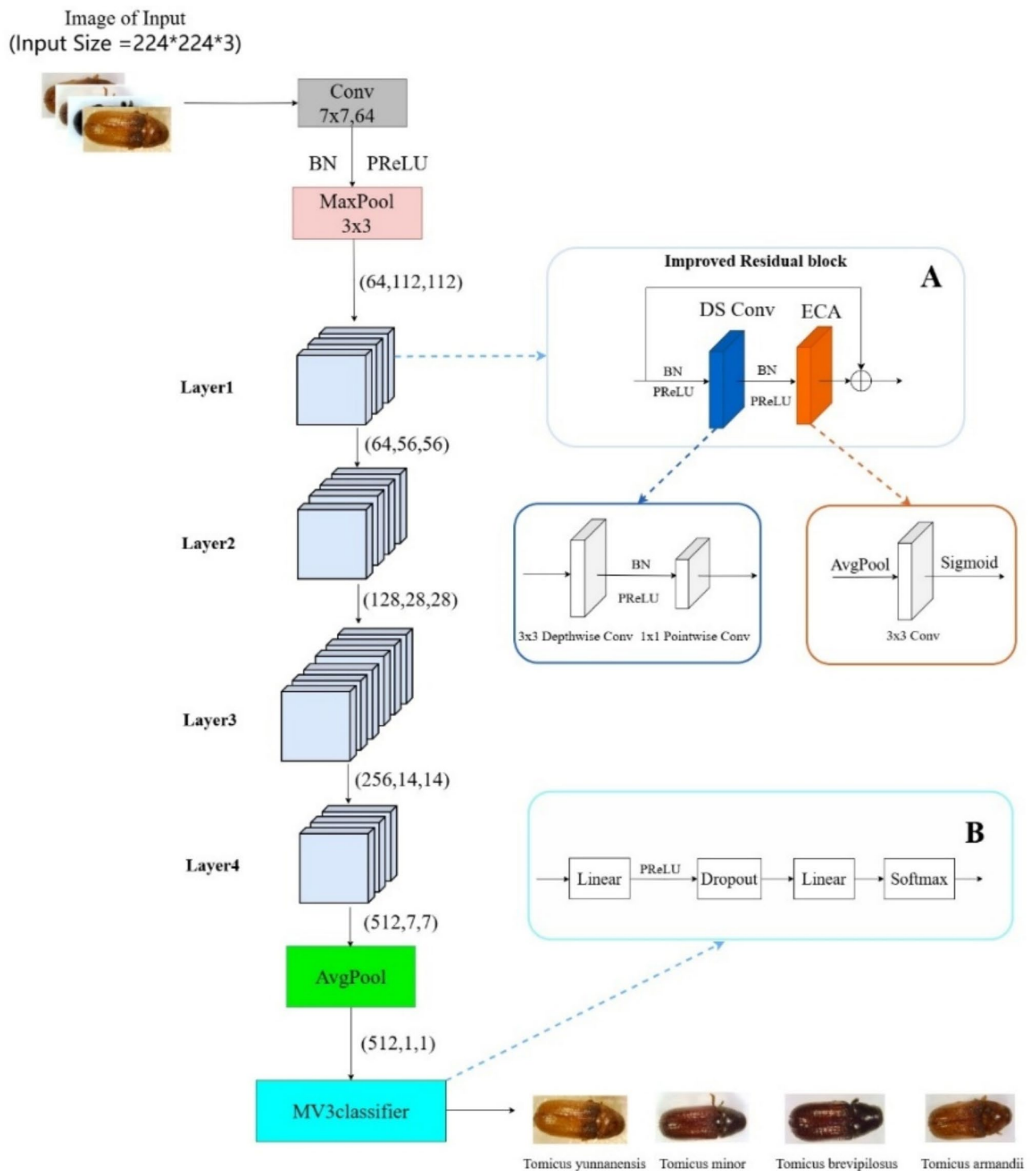
Fig. 4. Structure of DEMNet. (**A**) Improved Residual Block, Including DS convolution and ECA attention mechanism, the features within each channel are first processed and extracted through depthwise separable (DS) convolution, and then further refined by the ECA attention mechanism to enhance feature selection capability. (**B**) MobileNetV3 classifier, the input features are processed for classification through a sequence of layers, including a linear layer, a PReLU activation function layer, a dropout layer, another linear layer, and a SoftMax layer. (**C**) The ReLU activation function is entirely replaced with the PReLU activation function.

adjusting the number of channels in each network layer, we significantly reduced the model's parameter count without compromising performance, thus achieving the lightweight goal of the model and making it more efficient and flexible for practical applications.

Experimental methods

The computer used for the experiment was conducted on an AMD R9-9950X processor and 192 GB of memory. The implementation is carried out using the PyTorch deep learning framework, and the experimental model is trained on an NVIDIA 4090 GPU. During the training process, the optimizer used is the Adaptive Moment Estimation (Adam) algorithm, with the loss function set to Cross Entropy Loss. The training was executed over 100 epochs, with a batch size of 32.

To enhance the model's training efficiency, this study conducted systematic tuning of the learning rate. We set several different learning rates and conducted comparative experiments on the ResNet50 model to determine the optimal learning rate parameter. The experimental results, shown in Fig. 5, indicate that when the learning rate is set to 8.10-4, the model achieves the highest classification accuracy, significantly outperforming the model's performance under other learning rate settings.

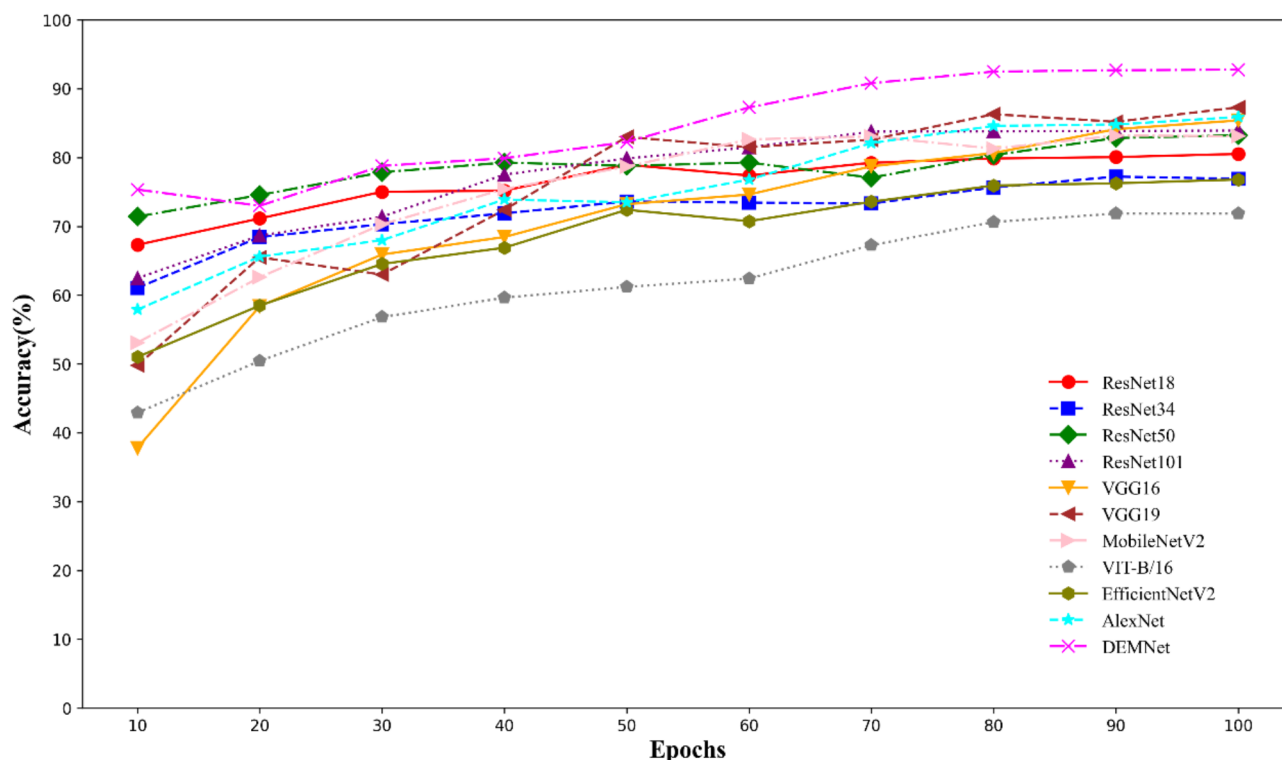### Results

Experimental results declaration

We conducted extensive comparative experiments on different enhanced modules to evaluate the performance of the model proposed in this study. First, we compared the method proposed in this study with traditional
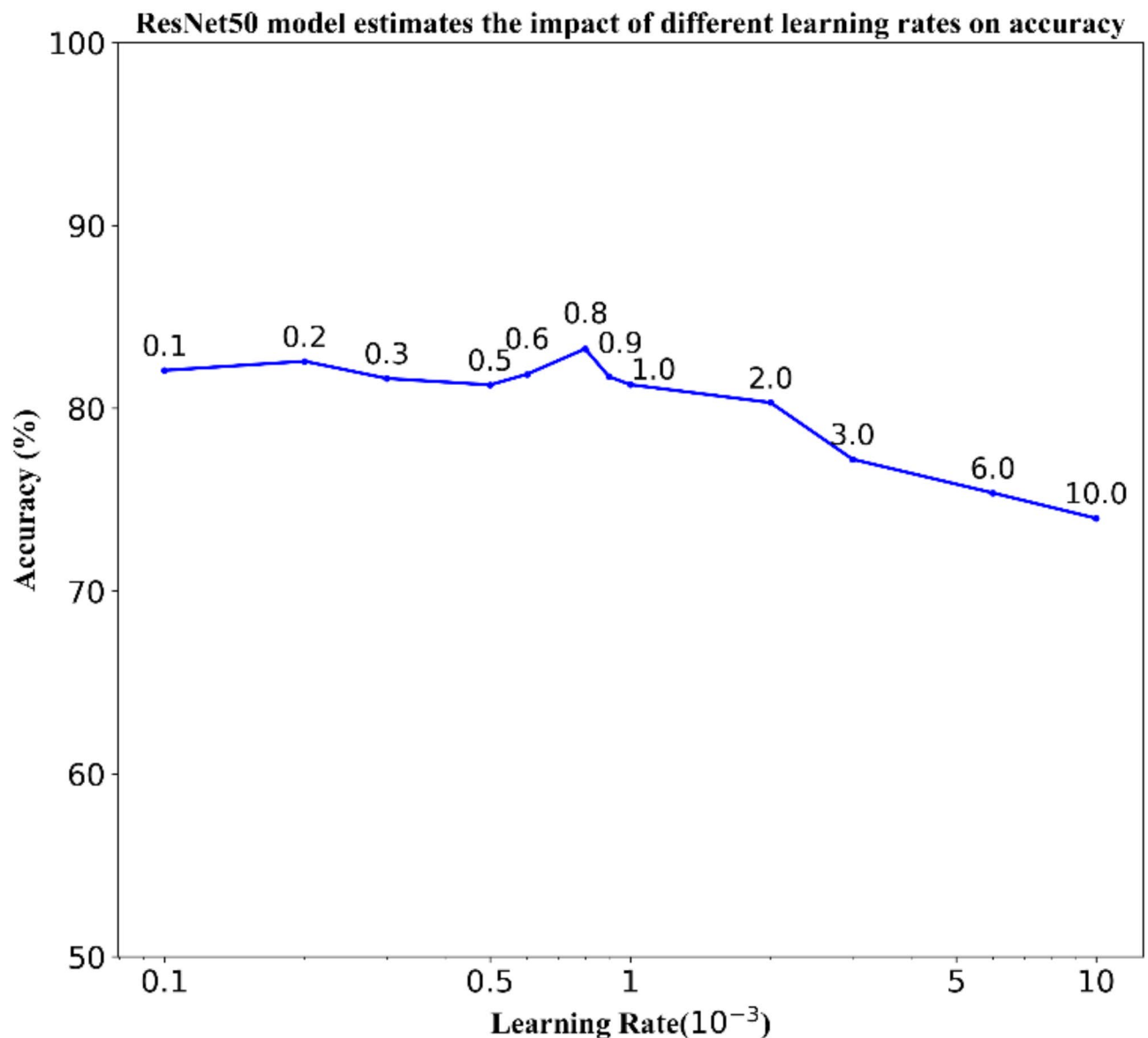
Fig. 5. Comparison of the effectiveness of different learning rates based on ResNet50 model.

Model	Accuracy	Loss	Params	Flops
ResNet18	80.5%	0.517	11.7 M	1.8G
ResNet34	77.2%	0.562	21.8 M	3.7G
ResNet50	83.3%	0.442	25.6 M	4.1G
ResNet101	83.9%	0.426	44.6 M	7.9G
MobileNetV2	83.2%	0.441	3.5 M	0.3G
VGG16	85.4%	0.393	138.4 M	15.5G
VGG19	81.8%	0.513	143.7 M	19.6G
VIT-B/16	71.9%	0.891	86.6 M	16.9G
EffientNetV2	76.8%	0.669	21.5 M	2.9G
AlexNet	85.9%	0.357	57.0 M	0.7G
DEMNet	**92.8%**	**0.220**	**1.6M**	0.7G

Table 3. Comparison experiments of different models.

Fig. 6. Results of the accuracy comparison experiments of different models over epochs.

classification models, using standard evaluation metrics to assess whether the proposed model demonstrated superior performance. Then, through ablation experiments, we selectively removed or modified certain components of the model, by testing different combinations of improved modules, comparing alternative attention mechanisms, and evaluating the effects of replacing activation functions. This approach allowed us to observe the impact on the performance of the network model and identify the optimal improvement strategy.

Comparison experiments of different models

To thoroughly validate the advantages of the proposed network model in terms of classification accuracy and lightweight design, we assessed it using several key performance indicators, including model accuracy, loss values, the number of training parameters, and floating-point operations per second (FLOPs). These metrics collectively provide a comprehensive view of the model's classification capabilities as well as its computational efficiency and resource usage in practical applications. To further validate the model's performance, during the training process, the learning rate and number of iterations (training epochs) were kept constant across all models. The new model was compared with ten classic image classification models, including ResNet18, MobileNetV2, and AlexNet. The experimental results are presented in Table 3; Fig. 6, the new model achieves a substantial reduction in parameters while maintaining high accuracy, demonstrating an efficient lightweight

design. These results clearly indicate that the proposed model outperforms existing classic models across multiple evaluation metrics, underscoring its broad application potential and advantages.

The comparative experimental results reveal that the proposed network model offers significant advantages in both classification accuracy and lightweight design. Firstly, in terms of classification accuracy, the new model achieved a final accuracy of 92.8%, which is 20.9% higher than the mainstream model VIT-B/16. For the loss metric, the new model reached a final loss value of 0.220, markedly lower than that of other models, indicating a faster and more stable convergence rate. Additionally, concerning lightweight design, the model demonstrates exceptional efficiency, containing only 1.6 M parameters, a 90% reduction compared to the original network. The model's FLOPs measure only 0.9G, significantly reducing computational overhead while maintaining high accuracy, thereby achieving the study's objective of lightweight design.

To verify that our model achieves fast and efficient image classification while maintaining high accuracy, we developed a *Tomicus* classification system on an embedded device using Python and Qt. The system deployed multiple pre-trained traditional models, including classic convolutional neural networks (e.g., the ResNet and VGG series), lightweight models (e.g., MobileNetV2), Vision Transformers (VIT-B/16), and EfficientNet, as well as the new model proposed in this study. By classifying the same *T.minor* image using different models, we obtained experimental results, presented in Table 4; Fig. 7. Our model not only demonstrates outstanding performance in classification accuracy but also achieves rapid processing with a time of only 0.1193 s.

Ablation experiment

To evaluate the influence of different attention mechanisms, the PReLU activation function, and various enhancement modules on the model's classification accuracy and parameter count, we conducted ablation experiments based on ResNet50. First, while keeping other enhancement modules constant, we compared the effects of different attention mechanisms and activation functions on model performance. This approach confirms the effectiveness of our research method in improving classification accuracy. Subsequently, we integrated DS convolutions with other enhancement modules and adjusted the output channel count, as shown in the layer configuration of the network structure diagram in Fig. 4, to achieve a model lightweight.

*The impact of different attention mechanisms on network performance*

In modern computer vision tasks, the efficient extraction of image features is paramount. To enhance this capability further, this experiment investigated the application of various attention mechanisms within the proposed network model. Keeping other modules constant, including DS convolution, the PReLU activation function, and the MobileNetV3 classifier—we compared five different attention mechanisms: Efficient Channel Attention (ECA), Squeeze-and-Excitation (SE), Convolutional Block Attention Module (CBAM), Mixed Local Channel Attention (MLCA), and Multi-Scale Channel Attention Mechanism (MSCAM). As shown in Table 5, the introduction of these attention mechanisms significantly impacted classification accuracy. Among them, the ECA module stood out, achieving a classification accuracy of 92.8%, which underscores its strong capability to capture essential features and its clear advantage in enhancing classification performance. The ECA mechanism improved the feature extraction capability of the *T.minor* classification model, allowing it to identify image regions with distinct feature differences more accurately, thereby improving classification accuracy.

The impact of prelu activation function on network performance

During the training process, selecting an appropriate activation function is essential to optimizing model performance. The PReLU activation function, as an enhancement of ReLU, introduces a learnable parameter that adjusts the output of negative values, providing greater flexibility and expressiveness. This modification allows PReLU to outperform ReLU in many scenarios. As shown in Table 6, by globally replacing the ReLU activation function in the original model with PReLU—while keeping other modules unchanged, such as DS convolution, ECA, and the MobileNetV3 classifier—the classification accuracy increased by 5.0%. Furthermore, compared to substituting ReLU with the Swish activation function, the global implementation of PReLU improved accuracy by an additional 3.3%. Thus, the PReLU activation function substantially enhances the performance of the proposed network model in classifying *Tomicus*. Notably, PReLU excels at capturing subtle feature variations,

Model	Probability	Times (in seconds)
ResNet18	81.3%	0.1890
ResNet34	78.1%	0.1502
ResNet50	81.7%	0.1371
ResNet101	85.0%	0.1456
MobileNetV2	82.0%	0.1095
VGG16	90.5%	0.3725
VGG19	86.2%	0.3982
VIT-B/16	68.6%	0.1749
EfficientNetV2	89.4%	0.2466
AlexNet	87.9%	0.1568
DEMNet	93.7%	0.1193

Table 4. Performance analysis of different models for detecting an image.
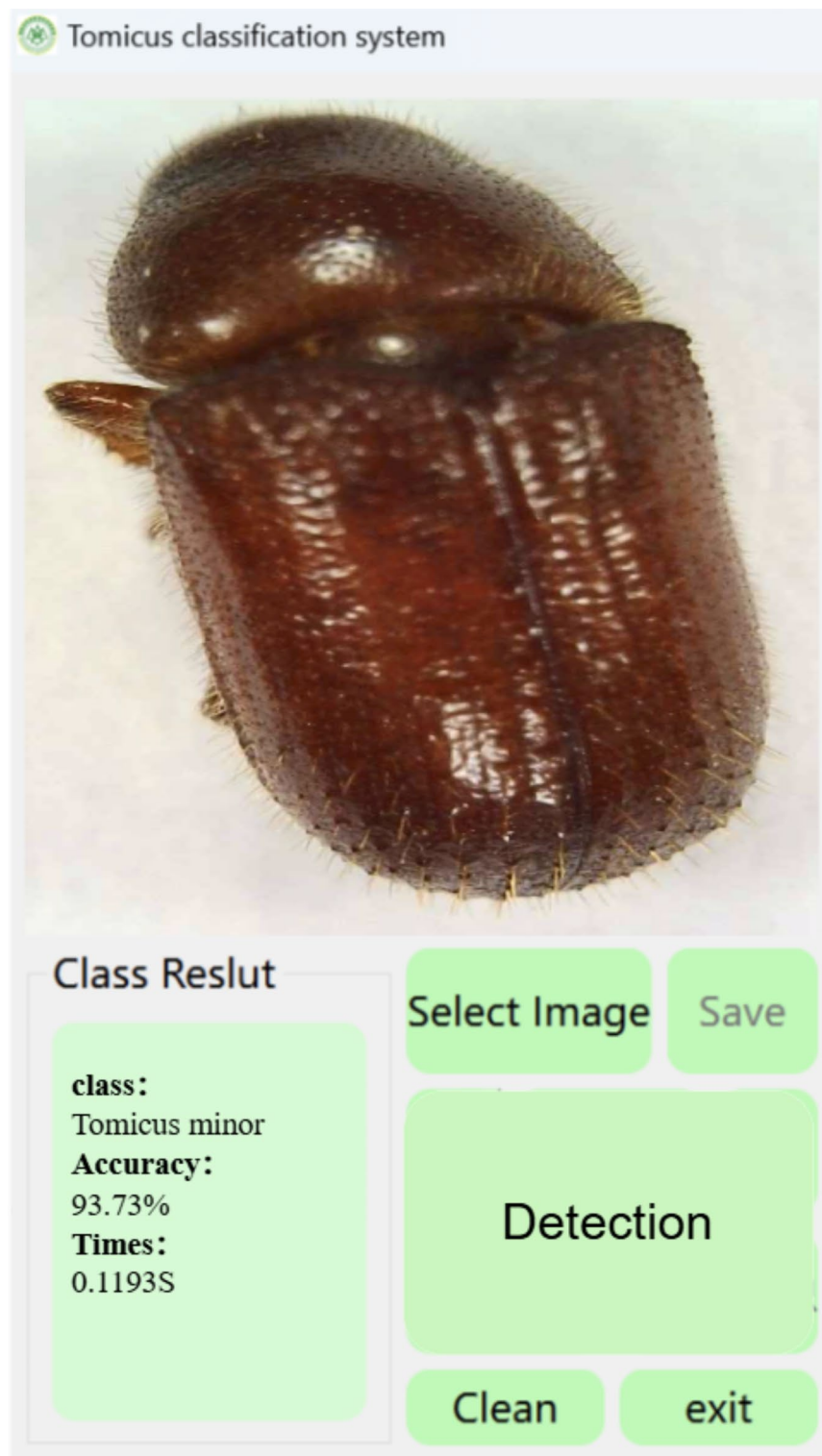
Fig. 7. The DEMNet model developed in this study demonstrates *Tomicus* classification performance on embedded devices.

Num	ResNet50	Other modules	Attention mechanism					Accuracy	Params
			ECA	SE	CBAM	MLCA	MSCAM		
1	✓	✓	✓					**92.8%**	**1.6M**
2	✓	✓		✓				91.4%	1.5 M
3	✓	✓			✓			91.4%	2.7 M
4	✓	✓				✓		90.1%	1.7 M
5	✓	✓					✓	89.7%	1.9 M

Table 5. The impact of different attention mechanisms on network performance.

Num	ResNet50	Other Modules	Activation Function			Accuracy
			ReLU	PReLU	Swish	
1	✓	✓	✓			87.8%
2	✓	✓		✓		**92.8%**
3	✓	✓			✓	89.5%

Table 6. The impact of different activation functions on network performance.

Num	ResNet50	PReLU	Improved Residual Block		MobileNetV3Classifier	Accuracy	Params
			DS	ECA			
1	✓					83.3%	25.6 M
2	✓	✓	✓			85.4%	1.3 M
3	✓	✓		✓		90.2%	9.8 M
4	✓	✓			✓	89.1%	10.4 M
5	✓	✓	✓	✓		90.7%	4.9 M
6	✓	✓	✓		✓	91.7%	1.4 M
7	✓	✓		✓	✓	91.5%	10.3 M
8	✓	✓	✓	✓	✓	**92.8%**	**1.6M**

**Table 7**. The impact of different improvement modules on the performance of network models.

including the distribution of notches on the *Tomicus* dorsum and changes in setae length, which significantly strengthens the model's ability to discern fine interclass differences. Consequently, PReLU enhances both the model's generalization capacity and its classification accuracy.

The impact of different improvement modules on network performance
As shown in Table 7, the impact of different improvement modules on the performance of the ResNet50 model is highly significant. The baseline ResNet50 model achieves an accuracy of 83.3% with 25.6 M parameters. With the addition of DS Convolution, parameter count is markedly reduced to 1.3 M while accuracy rises to 85.4%, demonstrating DS Convolution's effectiveness in reducing model complexity and boosting accuracy. Incorporating the ECA attention mechanism further improves accuracy to 90.2%, albeit with an increase in parameter count to 9.8 M. This indicates an accuracy gain but with added complexity. However, when DS Convolution is combined with ECA, accuracy rises slightly to 90.7% while parameters reduce to 4.9 M, achieving a balance between performance improvement and model efficiency. Adding the MobileNetV3 classifier yields an accuracy of 89.1% with a moderate parameter increase to 10.4 M, indicating a positive but incremental impact on performance. Integrating all modules yields the highest accuracy of 92.8% with only 1.6 M parameters, representing a 9.5% accuracy improvement over the baseline and a substantial reduction of 24 M parameters. These findings confirm that the strategic combination of modules significantly enhances classification accuracy while reducing parameter count and resource demands, underscoring each module's complementary role in improving model performance and efficiency.

Model performance evaluation
To thoroughly assess the classification performance of the proposed model on the *Tomicus* dataset, it is crucial to evaluate not only its overall accuracy but also its effectiveness in distinguishing between different *Tomicus* species. For this purpose, we employed a confusion matrix as a key evaluation tool to measure the model's performance across the four *Tomicus* species. The results, depicted in Fig. 8, provide a detailed visual representation of the model's predictions. The "Predicted Class" reflects the model's forecast category, while the "Actual Class" corresponds to the true species classification. This visualization allows for an intuitive assessment
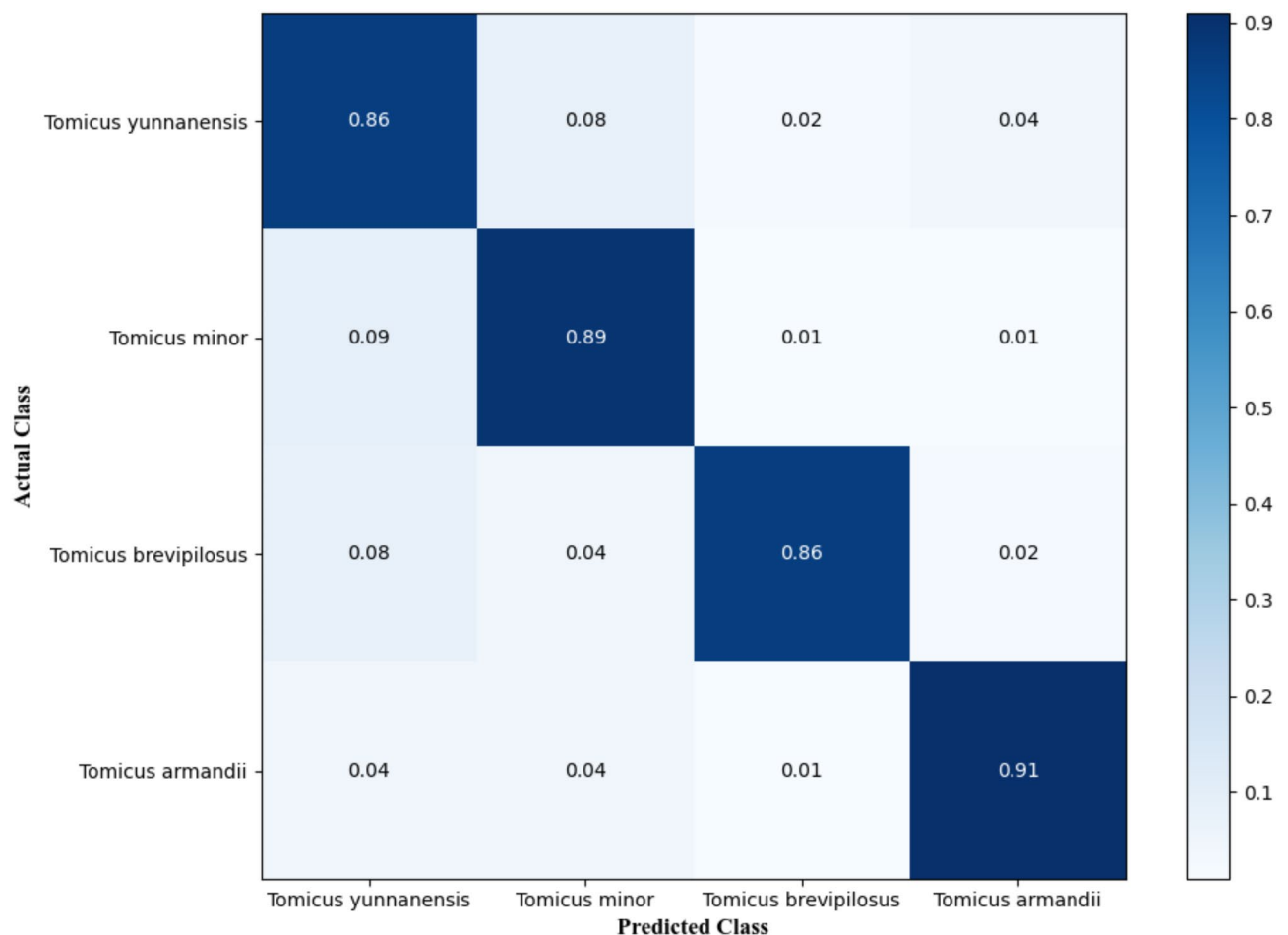
Fig. 8. Visualized result of confusion matrix.

of the model's classification precision for each category, as well as the extent of misclassifications across different species. The model demonstrates a high level of accuracy in correctly identifying these four species, with each class exhibiting a correct classification probability of over 85% and a relatively low rate of misclassification.

Discussion
Comparison of improvement results
In this study, we introduced several innovative enhancements to the ResNet50 model by integrating DS convolution, the ECA attention mechanism, and the PReLU activation function, while replacing the fully connected layer with the classifier structure from MobileNetV3. Experimental results revealed that the improved model not only achieved a high classification accuracy of 92.8% but also substantially reduced the parameter count to 1.6 M. To comprehensively evaluate the performance of the new model, we selected a range of classic architectures, from lightweight to large-scale networks, including AlexNet, VGG16, and MobileNetV2, as benchmarks in comparative experiments. This multidimensional evaluation demonstrated that our improved model, compared to traditional networks, significantly reduces the computational load while enhancing classification accuracy. It effectively achieves an optimal balance between lightweight design and superior classification performance. The proposed improvement strategies proved highly effective for the classification of *Tomicus*, successfully fulfilling the dual objectives of lightweight design and improved accuracy, highlighting the model's substantial potential for practical application.

Analysis of improvement modules
Optimizing and improving models often comes at the cost of increased computational complexity. However, the effectiveness of DS convolution in lightweight network design has been widely demonstrated. Sandler et al.[15] first introduced DS convolution in MobileNetV2, which drastically reduced computational burden. In this study, we adopted a similar approach by replacing the standard convolutions in ResNet50 with DS convolution, achieving a substantial 90% reduction in computational load compared to the original network, a finding that corroborates Sandler et al.'s results. DS convolution reduces complexity by decomposing standard convolution into two operations: depthwise convolution and pointwise convolution. This method effectively minimizes cross-channel computation, contributing to the model's lightweight design. However, while DS convolutions

efficiently reduce computational overhead, their impact on improving classification accuracy is limited. In contrast, attention mechanisms have shown significant potential in boosting classification accuracy. Due to the small size and subtle morphological features of *Tomicus* species, traditional neural networks struggle to achieve precise classification. To address this, we supplemented the DS convolution in ResNet50 with the ECA mechanism. This approach differs from the method proposed by Du et al.[32], who integrated the CBAM into ResNet50 for cotton seed quality detection. While their method improved detection accuracy, it also increased computational complexity. In comparison, our use of the ECA mechanism maintained a low parameter count while achieving a classification accuracy of 90.7% in identifying *Tomicus* species, an 8.2% improvement over the original ResNet50 model. The morphological similarities among the four *Tomicus* species, especially in terms of color, texture, and size, make classification challenging. The ECA mechanism enhances the model's attention to subtle differences, such as the dorsal setae length that distinguishes species like *T.brevipilosus* and *T.yunnanensis*. By refining inter-channel interactions, the ECA module significantly improves classification performance for morphologically similar species. Despite surpassing 90.0% classification accuracy, some misclassification persists due to the extreme similarity among *Tomicus* species. To further address this, we integrated the PReLU activation function proposed by He et al.[33], which has shown promise in complex image classification tasks. Additionally, we replaced the fully connected layer with the efficient MobileNetV3 classifier, as recommended by Howard et al.[21] for lightweight tasks. By combining PReLU's flexible handling of negative features with the efficiency of the MobileNetV3 classifier, we enhanced the model's nonlinear expressive capabilities and improved classification boundaries. This led to a noticeable reduction in the misclassification rate of *Tomicus* species, thereby achieving more accurate predictions. Our experimental results validate the effectiveness of the approaches proposed by He et al.[33] and Howard et al.[21] in the context of our model.

From a theoretical perspective, this study makes notable advancements in lightweight network design, focusing on reducing computational complexity while simultaneously improving classification accuracy. By integrating DS convolution, the ECA mechanism, the PReLU activation function, and the MobileNetV3 classifier, we present a novel lightweight model specifically designed for resource-constrained applications, such as pest detection and classification tasks on mobile devices or embedded systems. The primary innovation of this study lies in the comprehensive optimization of the ResNet50 architecture, achieving the dual objectives of enhanced classification accuracy and reduced model complexity. The experimental results demonstrate that DEMNet not only significantly improves classification accuracy but also drastically reduces the number of model parameters, validating the effectiveness of the proposed approach. This research provides robust theoretical support and practical insights for future lightweight model designs, with implications for a wide range of applications in image classification, particularly in environments with limited computational resources. We believe that this study offers promising prospects and substantial practical value in advancing lightweight networks in resource-constrained scenarios.

Future work

Despite its achievements, this study has certain limitations. While DEMNet shows impressive performance on the *Tomicus* classification dataset, its generalizability to other domains necessitates further validation. Future research could investigate the application of this model across a wider array of image classification tasks, including natural scene images and medical image classification, to evaluate its versatility. In our forthcoming work, we aim to address several challenges. First, given that the training data were captured against simple backgrounds, we seek to enhance the model's classification accuracy for *Tomicus* in more complex environments. To this end, we plan to collect samples from diverse backgrounds to broaden the training dataset and improve the model's performance. Second, this study concentrated solely on four *Tomicus* species from Yunnan Province. Future research should expand the range of *Tomicus* species and the sample dataset to bolster the model's generalization capabilities. Third, some of the models employed in the comparative experiments are relatively outdated. Thus, we will explore the use of more recent models, such as DenseNet[34], ResNeXt[35], and YOLOv10[36], for further comparative analyses.

Conclusion

This study successfully achieved a lightweight design and significantly improved classification accuracy through multiple optimizations to the ResNet50 architecture. Firstly, we replaced the standard convolutions in the residual modules with DS convolution, which greatly reduced the model's computational burden and parameter count. To enhance feature extraction capabilities, we integrated the ECA attention mechanism into each residual module while maintaining the skip connections. Additionally, we substituted the fully connected layer with the MobileNetV3 classifier, improving classification performance while preserving the lightweight structure. We also adjusted the output channels in each residual structure, thereby reducing both the parameter count and computational complexity. By replacing ReLU with PReLU, we accelerated model convergence and reduced the error rate. These optimizations led to a classification accuracy of 92.8% on the *Tomicus* dataset, with only 1.6 M parameters. Compared to the ResNet50 model, our method has improved accuracy by 9.5% compared to ResNet50, a 90% reduction in parameter count, a 3.5GMac reduction in FLOPs, and an inference speed of 0.1193 s per image, demonstrating a balanced performance in terms of speed, efficiency, and accuracy. In summary, we have successfully developed a high-speed, lightweight, and highly accurate deep learning model that provides an effective solution for the automatic identification of forestry pests, such as *Tomicus*, on embedded edge computing devices. This model provides essential technical support for monitoring forestry pests and eco-logical conservation, offering significant practical application value and valuable guidance for future related research.

Data availability

The datasets used and/or analyzed during the current study are available from the corresponding author on reasonable request.

References

1. Zhou, X. D., Jacobs, K., Morelet, M., Ye, H. & Wingfield, M. J. A new leptographium species associated with *Tomicus* piniperda in south-western China. *Mycoscience* **41**, 573–578 (2000).
2. HuiMin, W. et al. Differential patterns of ophiostomatoid fungal communities associated with three sympatric *Tomicus* species infesting Pines in south-western China, with a description of four new species. *MycoKeys* **50**, 93–133 (2019).
3. Lu, J., Zhao, T. & Ye, H. The Shoot-Feeding ecology of three *Tomicus* species in Yunnan Province, Southwestern China. *J. Insect Sci.* **14**, 1–10 (2014).
4. Wu, C., Chen, S., Yang, M. & Zhang, Z. Spatial distribution pattern and sampling plans for two sympatric *Tomicus* species infesting *Pinus yunnanensis* during the Shoot-Feeding phase. *Insects* **14**, 60 (2023).
5. Lu, T. T. et al. Comparative transcriptomics reveals the conservation and divergence of reproductive genes across three sympatric *Tomicus* bark beetles. *Comp. Biochem. Physiol. Part. D Genomics Proteom.* **49**, 101168 (2024).
6. Kirkendall, L. R., Faccoli, M. & Ye, H. U. I. Description of the Yunnan shoot borer, *Tomicus yunnanensis* Kirkendall & faccoli Sp. n.(Curculionidae, Scolytinae), an unusually aggressive pine shoot beetle from southern China, with a key to the Sp.cies of Tomicus. *Zootaxa* **1819**, 25–39 (2008).
7. Ye Hui, Y. H., Lu Jun, L. J. & Lieutier, F. On the bionomics of *Tomicus minor* (Hartig)(Coleoptera: Scolytidae) in Yunnan Province. (2004).
8. Chen, P., Lu, J., Haack, R. A. & Ye, H. Attack pattern and reproductive ecology of *Tomicus brevipilosus* (Coleoptera: Curculionidae) on *Pinus yunnanensis* in Southwestern China. *J. Insect Sci.* **15**, 43–43 (2015).
9. Li, X. Tomicus Armandii Li & Zhang (Curculionidae, Scolytinae), a new pine shoot borer from China. *Zootaxa* **2572**, 57–64 (2010).
10. Wu, C. X., Liu, F., Zhang, S. F., Kong, X. B. & Zhang, Z. Semiochemical regulation of the intraspecific and interspecific behavior of *Tomicus yunnanensis* and *Tomicus minor* during the Shoot-Feeding phase. *J. Chem. Ecol.* **45**, 227–240 (2019).
11. Wang, Y., Zhang, W., Gao, R., Jin, Z. & Wang, X. Recent advances in the application of deep learning methods to forestry. *Wood Sci. Technol.* **55**, 1171–1202 (2021).
12. Peng, H. et al. A lightweight crop pest classification method based on improved MobileNet-V2 model. *Agronomy* **14**, 1334 (2024).
13. He, K., Zhang, X., Ren, S. & Sun, J. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* 770–778 https://doi.org/10.48550/arXiv.1512.03385 (2016).
14. Li, Z. et al. Classification method of significant rice pests based on deep learning. *Agronomy* **12**, 2096 (2022).
15. Sandler, M., Howard, A., Zhu, M., Zhmoginov, A. & Chen, L. C. Mobilenetv2: Inverted residuals and linear bottlenecks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* 4510–4520 https://doi.org/10.48550/arXiv.1801.04381 (2018).
16. Howard, A. G. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *ArXiv Prepr*. https://doi.org/10.48550/arXiv.1704.04861 (2017). ArXiv170404861.
17. Kamal, K. C., Yin, Z., Wu, M. & Wu, Z. Depthwise separable Convolution architectures for plant disease classification. *Comput. Electron. Agric.* **165**, 104948 (2019).
18. Wang, Q. et al. ECA-Net: Efficient channel attention for deep convolutional neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* 11534–11542 https://doi.org/10.48550/arXiv.1910.03151 (2020).
19. Ni, H. et al. Classification of typical pests and diseases of rice based on the ECA attention mechanism. *Agriculture* **13**, 1066 (2023).
20. Guo, M. H. et al. Attention mechanisms in computer vision: A survey. *Comput. Vis. Media.* **8**, 331–368 (2022).
21. Howard, A. et al. Searching for mobilenetv3. In *Proceedings of the IEEE/CVF International Conference on Computer Vision* 1314–1324 https://doi.org/10.48550/arXiv.1905.02244 (2019).
22. Qin, X., Yang, G., Shao, Q., Zheng, H. & Zhang, M. Lightweight image matting algorithm based on deep learning. *IET Image Process.* **17**, 2829–2837 (2023).
23. Yan, C. et al. A lightweight network based on Multi-Scale asymmetric convolutional neural networks with attention mechanism for Ship-Radiated noise classification. *J. Mar. Sci. Eng.* **12**, 130 (2024).
24. Shaheed, K. et al. EfficientRMT-Net—An efficient ResNet-50 and vision Transformers approach for classifying potato plant leaf diseases. *Sensors* **23**, 9516 (2023).
25. Zheng, T. et al. An efficient mobile model for insect image classification in the field pest management. *Eng. Sci. Technol. Int. J.* **39**, 101335 (2023).
26. He, Z. et al. LiteNet: lightweight neural network for detecting arrhythmias at resource-constrained mobile devices. *Sensors* **18**, 1229 (2018).
27. Mi, Z., Zhang, X., Su, J., Han, D. & Su, B. Wheat Stripe rust grading by deep learning with attention mechanism and images from mobile devices. *Front. Plant. Sci.* **11**, 558126 (2020).
28. Zhang, Y., Zhan, Q. & Ma, Z. EfficientNet-ECA: A lightweight network based on efficient channel attention for class-imbalanced welding defects classification. *Adv. Eng. Inf.* **62**, 102737 (2024).
29. Ohn, I. & Kim, Y. Smooth function approximation by deep neural networks with general activation functions. *Entropy* **21**, 627 (2019).
30. Apicella, A., Donnarumma, F., Isgrò, F. & Prevete, R. A survey on modern trainable activation functions. *Neural Netw.* **138**, 14–32 (2021).
31. Wang, S. H. et al. Classification of Alzheimer's disease based on Eight-Layer convolutional neural network with leaky rectified linear unit and max pooling. *J. Med. Syst.* **42**, 85 (2018).
32. Du, X., Si, L., Li, P. & Yun, Z. A method for detecting the quality of cotton seeds based on an improved ResNet50 model. *Plos One*. **18**, e0273057 (2023).
33. He, K., Zhang, X., Ren, S. & Sun, J. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *Proceedings of the IEEE International Conference on Computer Vision* 1026–1034 https://doi.org/10.48550/arXiv.1502.01852 (2015).
34. Huang, G., Liu, Z., Van Der Maaten, L. & Weinberger, K. Q. Densely connected convolutional networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* 4700–4708 https://doi.org/10.48550/arXiv.1608.06993 (2017).
35. Xie, S., Girshick, R., Dollár, P., Tu, Z. & He, K. Aggregated residual transformations for deep neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* 1492–1500 https://doi.org/10.48550/arXiv.1611.05431 (2017).
36. Wang, A. et al. YOLOv10: Real-Time End-to-End Object Detection. https://doi.org/10.48550/arXiv.2405.14458 (2024).

Declarations

Competing interests
The authors declare no competing interests.

Additional information
Supplementary Information The online version contains supplementary material available at https://doi.org/10.1038/s41598-025-93407-5.

Correspondence and requests for materials should be addressed to Q.X. or Y.L.

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.