applied sciences

MDPI

Article

Wild Mushroom Classification Based on Improved MobileViT Deep Learning

Youju Peng [1], Yang Xu [1,2,*], Jin Shi [1] and Shiyi Jiang [1]

[1] College of Big Data and Information Engineering, Guizhou University, Guiyang 550025, China
[2] Guiyang Aluminum and Magnesium Design and Research Institute Co., Guiyang 550009, China
* Correspondence: xuy@gzu.edu.cn

Abstract: Wild mushrooms are not only tasty but also rich in nutritional value, but it is difficult for non-specialists to distinguish poisonous wild mushrooms accurately. Given the frequent occurrence of wild mushroom poisoning, we propose a new multidimensional feature fusion attention network (M-ViT) combining convolutional networks (ConvNets) and attention networks to compensate for the deficiency of pure ConvNets and pure attention networks. First, we introduced an attention mechanism Squeeze and Excitation (SE) module in the MobilenetV2 (MV2) structure of the network to enhance the representation of picture channels. Then, we designed a Multidimension Attention module (MDA) to guide the network to thoroughly learn and utilize local and global features through short connections. Moreover, using the Atrous Spatial Pyramid Pooling (ASPP) module to obtain longer distance relations, we fused the model features from different layers, and used the obtained joint features for wild mushroom classification. We validated the model on two datasets, mushroom and MO106, and the results showed that M-ViT performed the best on the two test datasets, with accurate dimensions of 96.21% and 91.83%, respectively. We compared the performance of our method with that of more advanced ConvNets and attention networks (Transformer), and our method achieved good results.

Keywords: attention mechanism; fine-grained; feature fusion; MobileViT

check for updates

Citation: Peng, Y.; Xu, Y.; Shi, J.; Jiang, S. Wild Mushroom Classification Based on Improved MobileViT Deep Learning. *Appl. Sci.* **2023**, *13*, 4680. https://doi.org/10.3390/app13084680

Academic Editor: Manuel Graña

Received: 3 February 2023
Revised: 29 March 2023
Accepted: 2 April 2023
Published: 7 April 2023

Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

Edible wild mushrooms are very nutritious, can improve the body's immunity, and have a variety of anti-disease abilities. In recent years, wild mushrooms are also frequently found in the market, and more and more people go to pick wild mushrooms. Wild mushrooms can be identified based on their shape, color, smell, microscopic features, pselaphesia, and other characteristics. However, due to the diversity of wild mushrooms, the challenges facing the classification of wild mushrooms are that the characteristics of wild mushrooms are small, and the image angle, illumination, target background, and other factors in the natural scene lead to significant differences between wild mushrooms, which are difficult to distinguish. Poisoning incidents occur from time to time, even leading to death. It is essential to identify the inedible and edible mushrooms correctly. This study aims to classify wild mushrooms using a method based on a combination of Transformer and ConvNets [1–3] to provide an auxiliary identification method for people.

In recent years, there have been several methods that can classify wild mushrooms, among which the most reliable method is molecular identification. However, this method is time-consuming, has a high threshold, and is unsuitable for real-time classification detection. Most of the current articles on wild mushroom classification use machine learning methods. Methods used in the literature include [4] decision tree (DT), plain Bayesian (NB), AdaBoost (AB), and support vector machine (SVM) machine learning algorithms. However, these methods rely on features that people manually label and the method learns the labeled wild mushroom information, rather than the collected visual information, for classification.

Since the feature attributes of wild mushrooms are closely related to the species, their ability to adapt and generalize is reduced in the absence of prior knowledge of the species. The accuracy rate plummets when visual information collected from images is passed into machine learning methods; thus, this method is inapplicable to the visual classification problem of wild mushrooms. Little research has been done to identify wild mushrooms using deep learning methods. In recent years, attempts have been made using ConvNets for the visual classification problem of wild mushrooms with good results [5]. Although ConvNets can extract local information on images well and the model converges quickly, there are limitations to extracting global information. The accuracy obtained when putting wild mushroom data into pure ConvNets is not high, so we believe the pure ConvNets model has some limitations for wild mushroom classification.

Since the ViT (Vision Transformer) model was proposed in 2020, the Transformer architecture has created a boom in the field of vision, and many people have introduced the attention mechanism into vision tasks. Transformer has the advantage of global extraction, but the number of its parameters is too large and lacks the inductive bias ability of convolution, which leads to overfitting. Thus, it requires pretraining on large-scale datasets and requires strong regularity [6–8]. The currently proposed method of wild mushroom classification does not fully consider the characteristics of minor differences between wild mushroom classes and significant differences within classes. As described above, a new global attention feature fusion network, M-ViT, is proposed, using local and global information interaction. For the problem of minor differences in the characteristics of wild mushrooms, multidimensional attention is fused for feature mapping of global high-level information (such as semantic information), which can help locate significant objects and help reduce the interference of the background. In addition, the convolution network is used to enhance the receptive field, extract local low-level information (such as texture), and improve the generalization ability of the wild mushroom classification model. Few studies have used deep learning methods to identify wild mushrooms, and there is a lack of interpretable analysis related to studying deep learning methods for wild mushrooms. This work was conducted to address these issues, and the main contributions of our article are as follows:

- The M-ViT network model was constructed and fine-tuned by adding an improved multidimensional attention module parallel to the MobileViT network. It enables the model to obtain a more effective interaction of local features and global feature information, more suitable for classifying wild mushroom datasets. A thorough search of the literature shows that this is the first study to use a combined Transformer and CNN model for the classification of wild mushrooms;
- The MV2 module in the original network combined with an improved attention mechanism for enhanced representation of important channels;
- For the "black box" problem of deep learning, we performed an interpretable analysis of the model, drawing the confusion matrix and heat map of the model.

We validated the effectiveness of the method on two datasets. The experimental results show that our method achieves advanced performance, providing an alternative to current CNN-based methods. Furthermore, our method improves the backbone model's performance (MobileViT) by 5.72% and 3.63% on the mushroom and MO106 datasets, respectively. Our tests and ideas provide helpful information for others, not only for specific cases of wild mushroom images.

1.1. Convolutional Networks (ConvNets)

Since the breakthrough of AlexNet [7] on vision problems, the main networks that followed are VGGNet [9], ResNet [10], MobileNet [11], EfficientNet [12], and RegNet [13], which are more profound and more effective. ConvNets have made a boom in vision tasks [7,12,14]. The authors of [15] compared three types of networks (AlexNet, VGGNet, and GoogLeNet). The best network accuracy of 82.63% was achieved on a small dataset of 1478 images to classify 38 wild mushroom species. The authors of [16] studied wild

mushroom image classification based on deep learning, whose dataset has only seven classes, and its coverage of wild mushroom species is small, which is not practical in use. The authors of [17] explored the classification and recognition of wild mushrooms based on Xception and ResNet-50 networks, whose datasets were mostly from Kaggle, and the existence of few categories and poor image quality or inclusion of noise led to lower generalization performance in practical application. The experimental process may not reach the index. The authors of [18] established a dataset of 23 categories of wild mushrooms by collecting data online. The accuracy also reached 92.17%, but its images are all from the Internet, the image resolution may not be uniform, the photo environment is noisy, the number is expanded mainly by rotation, and the detailed information is easily lost. Meanwhile, because of the minor inter-class differences and significant intra-class differences, the recognition of wild mushrooms belongs to fine-grained visual classification (FGVT), which is more complicated than the typical classification recognition problem with minor inter-class differences. The method used in [19] recognizes wild mushrooms based on the Inception-ResNet-v2 model from the perspective of fine-grained image classification, but does not consider the influence of the complex background of the images, and the recognition accuracy is relatively low. All of the above are implemented based on ConvNets methods, although they can extract local area information on images well. The model converges quickly, but there are limitations in extracting global information, so the accuracy of a pure ConvNet is low when wild mushroom data are used as input. The pure ConvNets model has some limitations for wild mushroom classification.

1.2. Vision Transformers

Transformer [20] debuted in 2017 as the first transformation model that relies entirely on self-attention to compute representations of inputs and outputs without using sequence-aligned recurrent neural networks or ConvNets for learning long sequences in NLP tasks. Inspired by the powerful representation capabilities of the transformer, the visual transformer (ViT) [21] was proposed in 2020 for computer vision tasks. This work divides images into multiple image patches (e.g., 16 × 16 pixels in size) and compares these image patches to tokens in NLP, which achieves, compared to ConvNets, image classification with impressive speed and accuracy. However, performing well requires a large-scale training dataset (i.e., JFT-300M). Subsequently, DeiT [22] introduced several training strategies to enable ViT to be effective using the smaller ImageNet-1K dataset. To consider the images' local and 2D nature, the Swin Transformer aggregates attention in a shifted window in a hierarchical structure with greater flexibility and generality. However, only partial local feature extraction is considered. Recently, Transformer has made some improvements on modules, including sparse attention [23–25], improved locality [6,26], pyramid design [27–29], and improved training strategies [22,30–32], which have improved model data efficiency to different degrees. Nevertheless, an extensive literature review reveals that Transformer-based models for wild mushroom classification are still being determined. The authors of [33] were the first to use the visual transformer large network (ViT-L/32) for wild mushroom classification. However, the dataset has only 11 classes, which could not be more generalizable. The number of parameters of the original ViT network is too large, and the network migration is complicated and not conducive to deployment and use. Improvements have been made to ordinary ViT [22,31,32], but none of these ViT variants can outperform SOTA-only convolutional models for ImageNet classification with the same amount of data and computation [34,35]. This suggests that the standard transformer layer may lack the inductive biases that ConvNets have. Because of the slight inter-class variation and significant intra-class variation in wild mushrooms, they belong with the matter of fine-grained visual recognition (FGVR), and native visual transformers cannot directly exploit their advantages on FGVR. For example, the perceptual field of ViT does not scale effectively because the length of patch tokens does not change with the increase in its encoder block. In addition, the model may not effectively capture the regional concerns carried in the patch tokens and is unsuitable for the fine-grained

classification of wild mushrooms. Combining the local extraction advantage of convolution with the global extraction advantage of attention, the hybrid design of transformer and convolutional layers will likely be a breakthrough for wild mushroom visual classification.

Moreover, this paper presents an improvement on the MobileViT [36] model, which combines the advantages of ConvNets and ViTs with the global modeling capability of Transformer and the inductive biasing capability of CNN. In this paper, the MobileViT block and the multi-axis attention module [37] in the original model are input side by side, and then their extracted information is fused, while making up for the lack of local information extraction in MobileViT to enable the model to obtain a more comprehensive and effective interaction of local features and global feature information. Also, in this paper, different training strategies are used to evaluate the model's performance.

2. Materials and Methods

2.1. Datasets

In our experiments, we mainly used two wild mushroom datasets, where each dataset was divided into training and validation sets according to a ratio of 8:2, and used for training and validation of the model to increase the generalization ability of the model. The sources of the wild mushroom datasets and their types are briefly described next.

2.1.1. *Mushroom*

These 261 wild mushrooms species were then divided into four types: deadly, toxic, conditionally edible, and edible; the group of conditionally edible mushrooms includes 8 different types of wild mushrooms species, deadly comprises 40 different types of wild mushrooms species, edible contains 67 different types of wild mushrooms species, and there are 146 different types of toxic wild mushroom species. Some of the wild mushroom images are shown in Figure 1.



Figure 1. Images of some species of mushrooms included in *Mushrooms*.

2.1.2. *MO106*

This dataset was constructed by cleaning and filtering two datasets, the 2018 FGVCx Fungal Classification Challenge dataset and the Wild Mushroom Observer website (MOW). *MO106* [38] consists of 29,100 images in 106 categories, with the largest class having 581 elements and the most minor 105 elements, while the mean value is 275. The image sizes ranged between 97 × 130 (smallest area) and 640 × 640 (largest area). Some of the wild mushroom images are shown in Figure 2.



Figure 2. Images of some species of mushrooms included in *MO106*.

2.2. Data Enhancement

Training a CNN requires much image data to prevent overfitting. Therefore, this paper enhances the input image data using Pytorch's built-in methods. We apply random cropping, random rotation, and random vertical and horizontal flipping to the image dataset for data enhancement during the training process. We apply the center crop to the images as a data enhancement operation during inference, and we apply the random order command to upset the order of all transform operations and increase the randomness of the operations. This is shown in Figure 3.



Figure 3. Images data augmentation.

2.3. M-ViT Model

Due to the various appearance features of wild mushrooms, some have great similarities. There are considerable differences between the same wild mushrooms due to shooting angle, light, and target background. Improving the recognition accuracy of fine-grained images requires the network model to be able to learn richer features and have more vital detail feature learning ability. Although Transformer currently achieves promising results on many classification tasks, the Transformer model is more challenging to optimize with many parameters and requires higher arithmetic power. The absolute position bias it uses leads to a more cumbersome migration to other tasks, which is not conducive to deployment. However, MobileViT is a lightweight, low-latency network that combines the advantages of CNN and ViTs network attention mechanisms. So, we chose MobileViT as the network's backbone and used a migration learning [39] approach. Our method uses the idea of splitting and fusion. First, we introduce the attention mechanism SE module in the MV2 structure of the Mobilenet network to enhance the representation of picture channels, then add the improved multidimensional attention module in the Block module, splice the features obtained from the attention module and the channels of the original feature map into 3C (3 channels), and finally, use the ASPP module to obtain a more long-distance relationship. The method uses dilated convolution to model, which performs multi-scale feature fusion on the input image. The structure of the M-ViT model is shown in Figure 4.

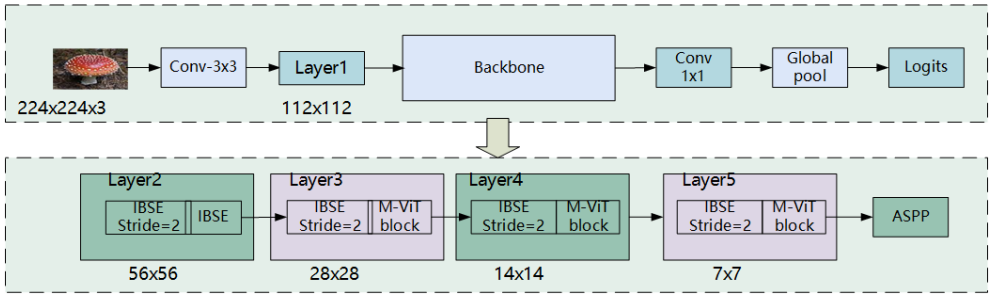


Figure 4. The M-ViT pipeline.

2.3.1. Inverted Bottleneck

The "inverted bottleneck" structure is adopted, in which the features undergo two steps of dimensionality enhancement and reduction during the convolution process. The inherent inductive bias of convolution improves the generalization ability and trainability of the model to a certain extent. In this paper, the SE attention mechanism module is added to the original residual structure to enhance the picture's channel characterization ability. The structure is shown in Figure 5.

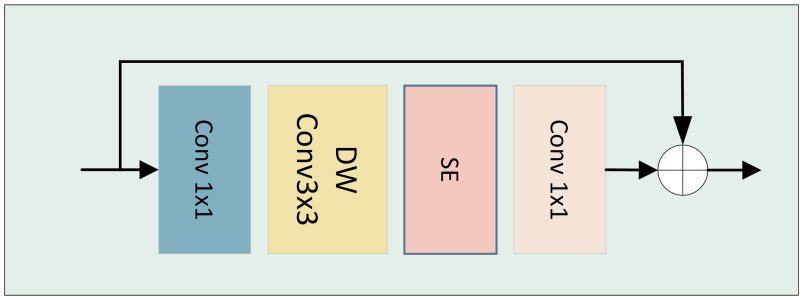


Figure 5. The MV2 Block.

In order to get a richer feature representation, the channel is first up-dimensioned using point-by-point convolution. Depthwise convolution is performed in the up-dimensioned projection space, followed by SE (squeeze excitation) for enhancing the representation of virtual channels. Finally, the dimensionality is recovered using point-by-point convolution again, as expressed in the following equation:

$$MV2(x) = x + Proj(SE(DWConv(Norm(x)))) \quad (1)$$

where $Proj$ and $Norm$ represent 1×1 convolution, and $DWConv$ is a convolution that reduces the number of parameters. Its input feature dimension is equal to the output feature dimension, and the kernel size is 3×3 in our experiment; SE is a common module. We do not go into details here.

2.3.2. M-ViT Block

As shown in Figure 6, we use both types of attention in parallel fusion to obtain more detailed local and global interaction information. We also use the typical structure of Transformer, gMLP [40], and common modules such as jump connection and LayerNorm. Since deep convolution can be considered conditional position encoding (CPE) [41], we introduce DW convolution in front of the attention module to replace the location encoding layer used in Transformer.

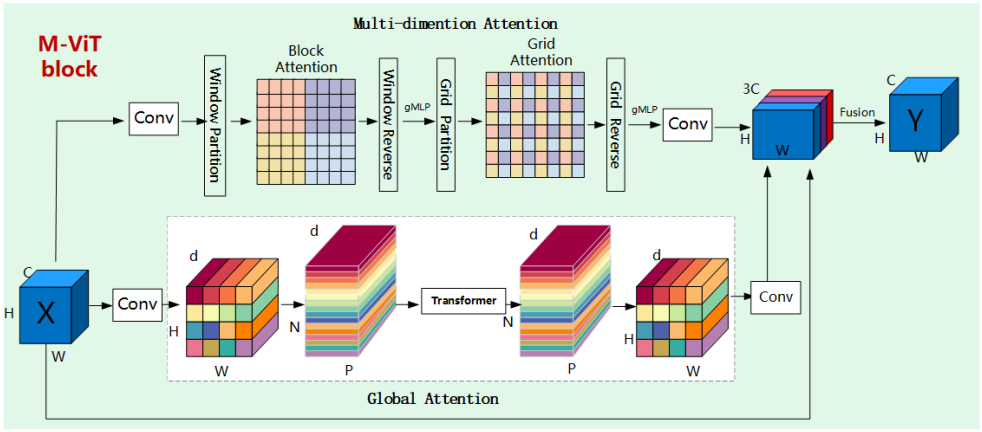


Figure 6. The M-ViT Block.

As shown in Figure 6, we input the feature map into the Multi-direction Attention and Global Attention modules to extract the local and global features of the image and then fuse the extracted features with the original feature map to obtain the feature map (3C) of three channels, and then restore the feature map through fusion (3×3 convolution). Where Conv is 1×1 convolution, the Window partition and Grid partition divide the window and grid into 4×4, block attention executes self-attention in the window, and grid attention globally focuses on the pixels of the entire window and grid space. The Global Attention module in the MobileViT model is insufficient for extracting local information only in the "local representations" section (including 3×3 convolution extraction of local features and 1×1 convolution adjustment channel number). Interested parties can read the original paper. Therefore, we have added the MDA module to enhance feature extraction. We introduce the MDA module in Section 2.3.4.

2.3.3. gMLP Block

It is well known that self-attention is the dynamic introduction of inductive bias by computing the spatial relationship between inputs. In contrast, gMLP is based on the MLP without the self-attention structure module, and only static parameterized channel projections and spatial projections exist. Compared with MLP, gMLP is less parametric and more effective. The structure is shown in Figure 7. Let the input of gMLP module $X \in R^{n \times d}$, n is the sequence length, and d is the feature dimension. The gMLP module is expressed as shown in the following equation:

$$gMLP(x) = x + proj(SGU(GELU(proj(norm(x))))) \quad (2)$$

$$SGU(x) = (proj(norm(split(x))) \quad (3)$$

where SGU is the acronym of spatial_gating_unit, the $proj$ stands for linear projection, $norm$ stands for regularization, and $gMLP$ is a commonly used module. Those interested in the details of the $gMLP$ module can read about it in the literature [40].

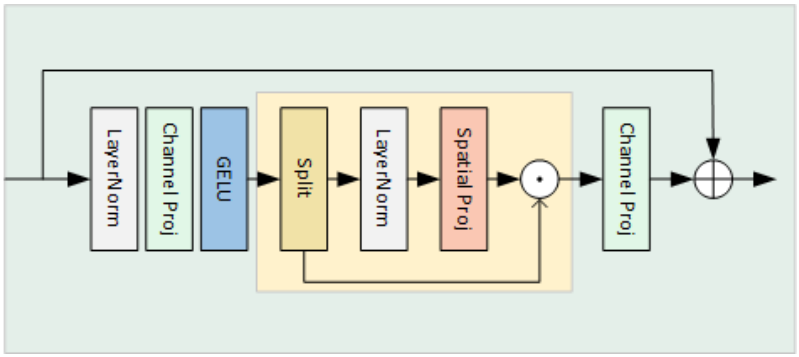


Figure 7. The gMLP Block.

2.3.4. Multidimension Attention

The multi-scale attention module decomposes the spatial axis into two sparse forms, local (block attention) and global (grid attention), to solve the quadratic complexity caused by global self-attention. Its structure is shown in Figure 8.

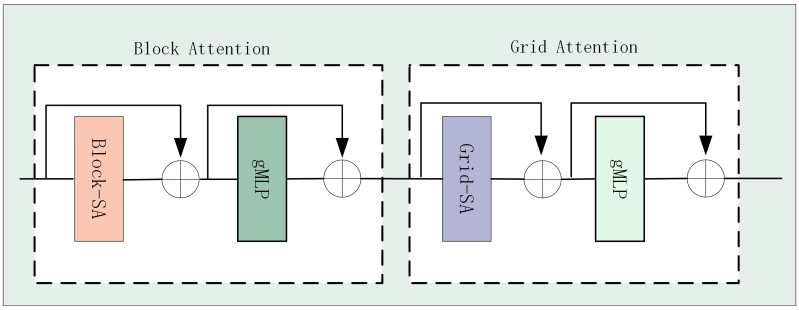


Figure 8. The MDA Block.

1. Block Attention: For the input feature map $X \in R^{H \times W \times C}$, it is transformed into a shape tensor $\left(\frac{H}{P} \times \frac{W}{P}, P \times P, C\right)$ to represent the division into non-overlapping windows, where the size of each window is $P \times P$, and finally, the RelAttention computation is performed in windows. First, we define the block $(\cdot)$ operation to convert the input feature graph into a non-overlapping window of $P \times P$ size, and then we define unblock $(\cdot)$ to do the opposite. The Block-SA Block in Figure 6 is to divide the Block into Windows and then performs the RelAttention operation, as in the following equations:

$$Block : (H, W, C) \rightarrow \left(\frac{H}{P} \times P, \frac{W}{P} \times P, C\right) \rightarrow \left(\frac{HW}{P^2}, P^2, C\right) \tag{4}$$

$$RelAttention(Q, K, V) = softmax(\frac{QK^T}{(\sqrt{d})})V \tag{5}$$

where Q, K, and V represent query, key, and value matrices, respectively, and d is the dim of Q and K.

$$Block\text{-}SA(x) = (x + Unblock(RelAttention(Block(DWConv(LN(x)))))) \tag{6}$$

$$x = x + gMLP(LN(x)) \tag{7}$$

2. Grid Attention: Grid attention uses Grid$(\cdot)$ to convert input features into $G \times G$ uniform grid to grid the input tensor as $\left(G \times G, \frac{H}{G} \times \frac{W}{G}, C\right)$ at which point the window of adaptive size is obtained $\frac{H}{G} \times \frac{W}{G}$, and finally on $G \times G$ The RelAttention calculation

is performed on the grid attention. Unlike the Block operation, we need additional transposes to place the grid dimension on the assumed spatial axis. Additionally, the Ungrid $(\cdot)$ is defined to perform the inverse operation to return it to 2D space, as in the follwoing equations:

$$\text{Grid} : (H, W, C) \rightarrow \left(G \times \frac{H}{G}, G \times \frac{W}{G}, C\right) \rightarrow \left(G^2, \frac{HW}{G^2}, C\right) \rightarrow \left(\frac{HW}{G^2}, G^2, C\right) \quad (8)$$

$$Grid\text{-}SA(x) = (x + UnGrid(RelAttention(Grid(DWConv(LN(x))))))) \quad (9)$$

$$x = x + gMLP(LN(x)) \quad (10)$$

3. Global Attention: For the input feature map $X \in R^{H \times W \times C}$, we first operate on the input with $DWConv$ and 1×1 convolution to obtain $X_L \in R^{H \times W \times d}$. DW convolution is used to learn the local and channel spatial information to prevent the loss of spatial information of channels, and 1×1 convolution is used to project the input features into the high-dimensional space. A visual transformer (ViT) with multi-headed self-attentiveness is used for modeling to obtain longer-distance relations. However, ViT has many parameters and weak optimization capability because ViT lacks inductive bias. To enable the Transformer (ViT) to learn a global representation with spatial inductive bias, first, the X_L is expanded into N non-overlapping patches $X_U \in R^{H \times N \times d}$, where $P = wh$, $N = \frac{HW}{P}$ is the number of patches and $h \leq n$, $w \leq n$ is the height and width of the patches. Each pixel in the patches $P \in \{1, \cdots, P\}$ is modeled by Transformer to get $X_G \in R^{P \times N \times d}$, as in the following equations:

$$X_G(p) = Transformer(X_U(P)), 1 \leq p \leq P \quad (11)$$

Unlike ViT, which loses the spatial order of pixels, global attention does not lose either the patch order or the spatial order of pixels within each patch. As shown above, first collapsing the $X_G \in R^{P \times N \times d}$ to obtain $X_F \in R^{H \times W \times d}$ and then using a 1×1 convolution to X_G adjustment to obtain the C dimensional features. Since $X_U(p)$ uses DW convolution to encode the local information, and $X_G(p)$ encodes the global information at the p-th position in the P patch, it is possible for the X_G to make sense of the X perception of the global information.

Finally, the M-ViT Block module uses the Fusion convolution module to fuse the local and global features extracted from the Multi-scale Attention module and the Global Attention module.

3. Results

In this section, we present a detailed analysis of the experiments, including the dataset, performance metrics, detailed results, and an analysis of the ablation experiments to better illustrate the effectiveness of the M-ViT architecture.

3.1. Implementation Details and Performance Evaluation Metrics

We implemented the M-ViT model with the Python 3.8 and PyTorch 1.7.0 framework. The input image size is $224 \times 224 \times 3$. Data enhancements, such as horizontal flipping, rotation, and random cropping, are used on the image to increase the data diversity. The proposed model is trained on a single high-performance NVIDIA GeForce GTX 3090 graphics card. Optimization is performed using the AdamW [42] optimizer with a training epoch of 300, a batch size of 32, a learning rate of 5×10^{-4}, and a weight decay of 5×10^{-3}. Our experiments use accuracy, precision, recall, and specificity rubrics to evaluate the model's performance.

1. Accuracy: This indicates the accuracy of the prediction result, the number of correctly predicted samples divided by the total number of samples, as in the following equation:

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN} \tag{12}$$

2. Precision: This indicates the probability of correctly predicting a positive sample among the predicted to be positive in the prediction result, as in the following equation:

$$Precision = \frac{TP}{TP + FP} \tag{13}$$

3. Recall: Also known as the sensitivity, this indicates the probability that a positive sample of the original sample will be correctly predicted as a positive sample in the end, as in the follwing equation:

$$Recall = \frac{TP}{TP + FN} \tag{14}$$

4. F1-Score: This represents the harmonic mean evaluation metric of precision and recall. The F1-score is a weighted average of the model's precision and recall, with a maximum value of 1 and a minimum value of 0, as in the follwing equation:

$$F1\text{-}Score = 2 * \frac{Precision * Recall}{Precision + Recall} \tag{15}$$

5. Specificity: This describes the proportion of identified negative samples to all negative samples, as in the following equation:

$$Specificity = \frac{TN}{FP + TN} \tag{16}$$

where TP (true positive) indicates the number of samples judged to be positive and are positive; FP (false positive) indicates the number of samples judged to be positive but are negative; TN (true negative) indicates the number of samples judged to be negative and are negative; and FN (false negative) indicates the number of samples judged to be negative but are positive. Negative indicates the number of samples judged to be negative but are positive.

3.2. Experimental Results and Analysis

We validated the effectiveness of our proposed model on two wild mushroom datasets for the visual classification task of wild mushrooms. We compared the results obtained by the method in this paper on the MO106 and mushroom datasets with the Mobilenet-V2, Efficienetb0, and ConvNeXt networks, which are typical convolutional networks, as well as the more recent advanced attention networks Transfomer and Swin-transformer networks. Table 1 compares the performance of the proposed method with the methods proposed in other published studies and shows that our method outperforms several other methods in terms of accuracy. As can be seen from the data in the table, our method outperforms MobileViT, and both our method and MobileViT network use images with a resolution of 224 as input. The accuracy obtained by this method on the mushroom dataset (91.83%) is 5.72% higher than that obtained by the base model backbone on this dataset (86.1%) and 11.59% higher than that obtained by ConvNeXt (80.24%), a variant of the current state-of-the-art convolutional neural network; the accuracy obtained on the MO106 dataset (96.21%) is 11.59% higher than that obtained by the base model backbone network on this dataset. The accuracy obtained on the MO106 dataset (96.21%) is 3.63% higher than that obtained by the base model backbone network on this dataset (92.58%). The method in this paper achieved good results. The method in this paper is based on the standard MobileViT network improvement, which has the potential for further improvement. Table 2

shows the classification performance of the M-ViT model on the mushroom dataset for each class of wild mushrooms. The results show that the scores of the M-ViT model achieved good results in each wild mushroom category. For edible and poisonous types of wild mushrooms, M-ViT models had higher F1-score of 94.9% and 92.9%, respectively. *Conditionally_edible* (87.3%) and *deadly* (81.9%) are less effective in wild mushrooms; this is because the number of mushrooms species in the *poisonous* group accounts for a relatively small proportion in the *Conditionally_edible* and *deadly* groups. The data imbalance in deep learning skews results in favor of those with more data, therefore, we conducted the experimental verification on the *MO106* dataset with relatively uniform data distribution, and there was no such difference. Thus, our model achieved good performance.

Table 1. Comparison with other methods on two datasets.

Method	Epoch	Batch Size	Accuracy (%)		Loss	
			MO106	Mushroom	MO106	Mushroom
Resnet34	300	32	87.44	75.0	0.43	0.178
Mobilenet-V2	300	32	76.49	65.2	1.37	0.944
Efficienet	300	32	89.91	84.9	0.07	0.126
ConvNeXt	300	32	88.23	80.24	1.056	1.079
Transfomer	300	32	90.90	78.25	0.301	1.027
Swin-transformer	300	32	88.47	76.31	0.622	1.512
ShuffleNet	300	32	89.54	83.5	0.12	0.164
Backbone	300	32	92.58	86.11	0.603	0.90
Our method	300	32	96.21	91.83	1.053	0.871

Table 2. Classification evaluation results of M-ViT on the mushroom dataset.

Class	Precision (%)	Recall (%)	Specificity (%)	F1-Score (%)
Conditionally_edible	93.2	82.1	99.7	87.3
Deadly	90.4	74.8	98.7	81.9
Edible	97.6	92.3	99.1	94.9
Poisonous	89.3	96.8	86.8	92.9

3.3. Ablation Experiments

This paper proposes a Transformer network M-ViT based on multidimensional fine-grained attention information fusion with MobileViT as the base network. To address the problem that the attention module of the MobileViT network is not comprehensive for fine-grained feature extraction of wild mushrooms, firstly, we propose a local and Global Attention module MDA, which is input to the network side by side with the attention module of the backbone network for complete feature extraction of local and global features; secondly, an attention SE module is introduced into the residual module in MobileViT for enhancing the representation of the channel, and then a multi-scale information extraction module ASPP is introduced to feature maps at different scales are fused. Finally, different training strategies are used for training comparison to optimize the model's performance. In order to verify the effectiveness of each module in the model proposed in this paper, this section conducts ablation experiments on two datasets, MO106 and mushroom, and compares them with the backbone network.

First, the function of the MDA module is verified, and then the different arrangements of the MDA module are added to Layer2, Layer3, and Layer4 based on the MobileViT network. As shown in Table 3, "$\surd$" indicates that the MDA module is added to the corresponding layer in the MobileViT network, and "×" indicates that it is not added, but the original attention block is used.

Table 3. The effect of Multidimension Attention on the model at different locations.

		Layer2	Layer3	Layer4	Dataset	Accuracy (%)	Dataset	Accuracy (%)
Backbone		×	×	×	MO106	92.58	mushroom	86.11
Ours	1	√	×	×	MO106	94.74	mushroom	90.13
	2	×	√	×	MO106	94.69	mushroom	90.2
	3	×	×	√	MO106	95.04	mushroom	90.34
	4	√	√	×	MO106	87.53	mushroom	79.54
	5	×	√	√	MO106	89.41	mushroom	82.63
	6	√	√	√	MO106	83.59	mushroom	75.6

Table 3 shows the seven cases of the original MobileViT model with MDA modules added to different layers. On the MO106 dataset, the accuracy of adding the MDA module to Layer2, Layer3, and Layer4 is 94.74%, 94.69%, and 95.04%, respectively, which is 2.16%, 2.11%, and 2.46% higher compared with the backbone. On the mushroom dataset, the accuracy of adding the MDA module to Layer2, Layer3, and Layer4 was 90.13%, 90.20%, and 90.34%, respectively, which improved by 4.02%, 4.09% and 4.23%, respectively, compared with the backbone. However, after adding two or three MDA modules at different locations in Layer2, Layer3, and Layer4 simultaneously, the model's accuracy decreased, which was affected by overfitting after the model became complex. Therefore, considering the accuracy and the number of network parameters, this study chose to add MDA at Layer 4 for the next improvement. We then investigated the effect of different MLPs in the MDA module on the network, and the results are shown in Table 4.

Table 4. Effect of different MLPs on the model at different locations.

Block	Dataset	Accuracy (%)	Dataset	Accuracy (%)
MLP	MO106	95.04	mushroom	90.34
gMLP	MO106	96.15	mushroom	91.42
GluMLP	MO106	87.11	mushroom	90.00

Table 4 shows the effect of adding different MLP modules to the model. From the experimental results, the gMLP module improved both datasets' accuracy, so we chose to include the gMLP module.

Table 5 shows the effect of having the SE module on the model. The SE module is an attention mechanism that enhances the channels' representational ability. In contrast, our MDA module is an attention module that blends global and local attention. After adding the SE module, the accuracy of the two datasets improved a little, indicating that the SE module impacts the characterization ability of the mushroom access channel.

Table 5. Impact of SE module on the model.

SE	Dataset	Accuracy (%)	Dataset	Accuracy (%)
√	MO106	95.47	mushroom	91.37
×	MO106	96.15	mushroom	91.42

Table 6 shows the effect of the ASPP module on the model. We added the ASPP module after Layer5 of the model for fusing the different scale information of the model, and the ASPP module increased the perceptual field of the network without downsampling, which made up for the lack of the Transformer's perceptual field and enhanced the network's ability to obtain multi-scale contextual information. As shown in Table 6, the model's accuracy was improved to 91.83% on the mushroom dataset and 96.21% on the MO106 dataset after adding this module.

Table 6. Impact of the ASPP module on the model.

ASPP	Dataset	Accuracy (%)	Dataset	Accuracy (%)
√	MO106	96.21	mushroom	91.83
×	MO106	96.15	mushroom	91.42

3.4. Model Evaluation and Interpretability Analysis

In order to evaluate the effectiveness of the proposed network M-ViT, the model with the best training results was saved for testing. For the "black box" problem of deep learning, the models were analyzed for interpretability, and the models' confusion matrix and heat map were plotted.

3.4.1. Confusion Matrix

The confusion matrix is a common metric for judging model results and belongs to the model evaluation section. In order to evaluate the classification effect of the model on each category, we drew the confusion matrix for visual analysis of the mushroom dataset on the M-ViT model and other models, as shown in Figure 9. The ShuffleNet network performed worst in the category *conditionally_edible*, all models have better results in the edible category, and our model has higher accuracy than the other eight models in all four categories.

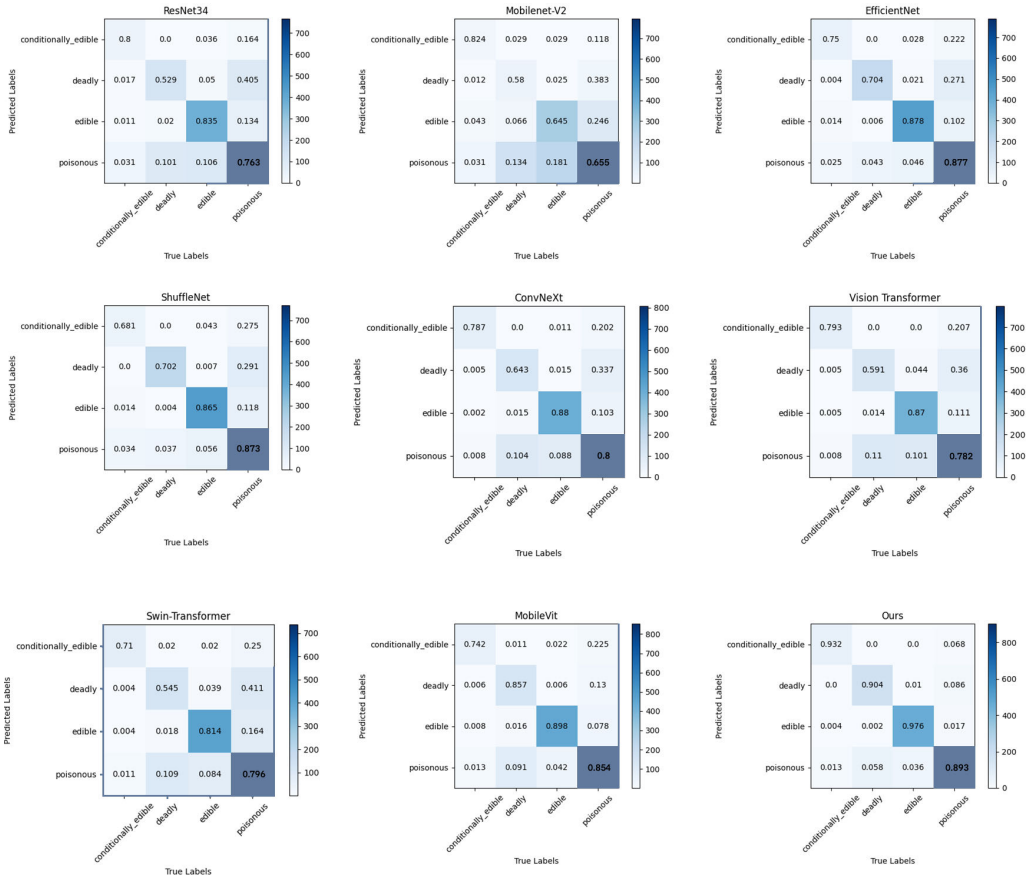


Figure 9. Confusion matrices of the classification results for the mushroom dataset of all models.

3.4.2. Grad-CAM

We used Grad-CAM [43] to draw the following heat map, which can be used to analyze whether the network learns the correct features or information, increasing the interpretability of the model. The formula is shown below in the following equations:

$$L^{c}_{Grad\text{-}CAM} = RELU\left(\sum_{k} \alpha^{c}_{k} A^{k}\right) \tag{17}$$

where A represents a specific feature layer, which in our model is the feature layer output from the last convolution layer; k represents the feature A of the k channel; c represents the category; and A^k represents the feature layer A in the channel k in the feature layer. α^c_k represents the data for the A^k.

The weights of the α^c_k in the calculation formula are as in the following equation:

$$\alpha^{c}_{k} = \frac{1}{Z}\sum_{i}\sum_{j}\frac{\partial y^{c}}{\partial A^{k}_{ij}} \tag{18}$$

where y^c represents the score predicted by the network for the category c; A^k_{ij} represents the feature layer A in the channel k with coordinates ij as the position of the data; and Z is equal to the width of the feature layer × height. We visualized the backbone model and our model drawing heat maps for the mushroom datasets for analysis, and the results are shown in Figures 10–12. It is clear from the image results that our model can focus more comprehensively on the characteristics of wild mushrooms .

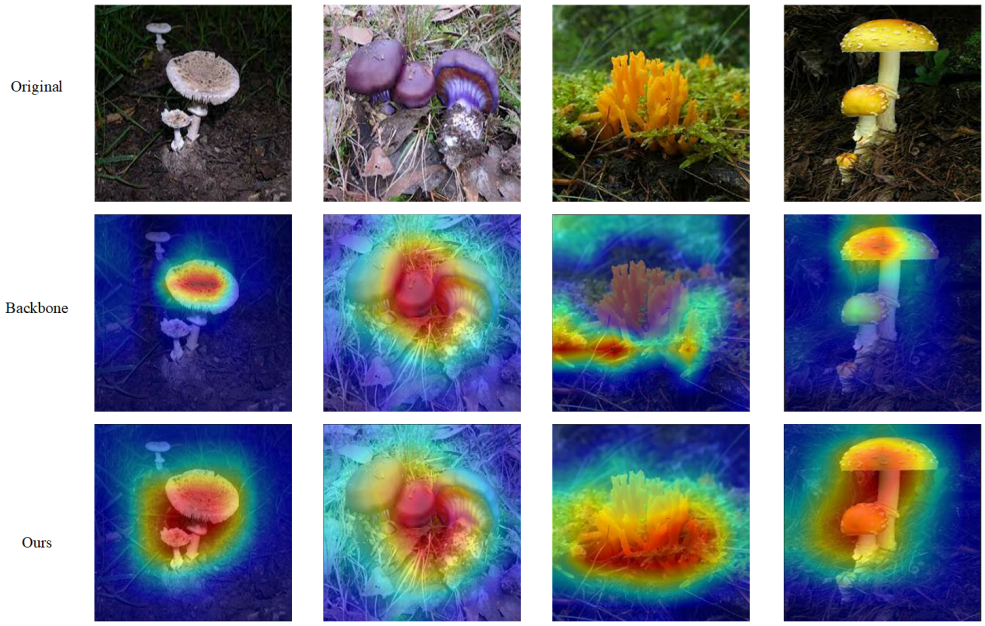


Figure 10. The thermodynamic diagrams of backbone model and ours.



Figure 11. The thermodynamic diagrams of backbone model and ours.



Figure 12. The thermodynamic diagrams of backbone model and ours.

4. Conclusions

In this paper, we propose an M-ViT model that increases the impact of fine-grained recognition by combining low-level features and high-level feature information of wild mushrooms. Our approach uses multidimensional attention to sample the planar and spatial information of wild mushroom images and decompose the spatial axis information to reduce the computational complexity of the attention operator for image representation extraction. Add the multidimensional and Global Attention modules to the process of extract potential representations, use ASPP fusion modules to fuse the extracted image information of different sizes, and improve the distinguish ability of possible image representations. In this study, MobileViT, a lightweight model combining convolution and Transformer, is introduced into the visual classification transformer model of wild mushrooms for the first time and achieves a better recognition rate with small model

parameters. Many experiments have been carried out using the *MO106* and *mushroom* datasets.Through experiments, our method was 91.83% accurate on the *mushroom* dataset and 96.21% on the *MO106* dataset, which is higher than the Mobilenet-V2, ConvNeXt, Transformer, and Swin-transformer networks; it can be used as an auxiliary identification method of wild mushrooms.

Author Contributions: Conceptualization, Y.P. and Y.X.; methodology, Y.P. and Y.X.; software, Y.P. and J.S.; validation, Y.P. and Y.X.; formal analysis, Y.P. and S.J.; investigation,Y.P. and S.J.; resources, Y.P.; data curation, Y.P., J.S. and S.J.; writing—original draft preparation, Y.P.; writing—review and editing, Y.P. and J.S.; visualization, Y.P.; supervision, Y.P. and Y.X.; funding acquisition, Y.X. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by Guizhou Provincial Key Technology R&D Program [2021] General 176.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The datasets generated and analyzed during the current study are available in the Mushroom repository and MO106 repository at https://www.kaggle.com/datasets/derekkunowilliams/mushrooms/ (accessed on 16 July 2022) and http://keplab.mik.uni-pannon.hu/images/mo106/ (accessed on 5 August 2022).

Acknowledgments: The authors would like to thank the editors and the anonymous reviewers for their valuable suggestions.

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

References

1. Liu, Q.; Fang, M.; Li, Y.; Gao, M. Deep learning based research on quality classification of shiitake mushrooms. *LWT* **2022**, *168*, 113902. [CrossRef]
2. Molina-Castillo, S.; Espinoza-Ortega, A.; Thomé-Ortiz, H.; Moctezuma-Pérez, S. Gastronomic diversity of wild edible mushrooms in the Mexican cuisine. *Int. J. Gastron. Food Sci.* **2023**, *31*, 100652. [CrossRef]
3. Ford, W.W. A new classification of mycetismus (mushroom poisoning). *J. Pharmacol. Exp. Ther.* **1926**, *29*, 305–309.
4. Tutuncu, K.; Cinar, I.; Kursun, R.; Koklu, M. Edible and poisonous mushrooms classification by machine learning algorithms. In Proceedings of the 2022 11th Mediterranean Conference on Embedded Computing (MECO), Budva, Montenegro, 7–10 June 2022; pp. 1–4.
5. Abdulnabi, A.H.; Wang, G.; Lu, J.; Jia, K. Multi-task CNN model for attribute prediction. *IEEE Trans. Multimed.* **2015**, *17*, 1949–1959. [CrossRef]
6. Han, K.; Xiao, A.; Wu, E.; Guo, J.; Xu, C.; Wang, Y. Transformer in transformer. *Adv. Neural Inf. Process. Syst.* **2021**, *34*, 15908–15919.
7. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. Imagenet classification with deep convolutional neural networks. *Commun. ACM* **2017**, *60*, 84–90. [CrossRef]
8. Guo, Y.; Zheng, Y.; Tan, M.; Chen, Q.; Li, Z.; Chen, J.; Zhao, P.; Huang, J. Towards accurate and compact architectures via neural architecture transformer. *IEEE Trans. Pattern Anal. Mach. Intell.* **2021**, *44*, 6501–6516. [CrossRef]
9. Simonyan, K.; Zisserman, A. Two-stream convolutional networks for action recognition in videos. In Proceedings of the Advances in Neural Information Processing Systems, Montreal, QC, Canada, 8–13 December 2014; Volume 27.
10. 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, Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.
11. Howard, A.G.; Zhu, M.; Chen, B.; Kalenichenko, D.; Wang, W.; Weyand, T.; Andreetto, M.; Adam, H. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv* **2017**, arXiv:1704.04861.
12. Tan, M.; Le, Q. Efficientnet: Rethinking model scaling for convolutional neural networks. In Proceedings of the International Conference on Machine Learning, Long Beach, CA, USA, 10–15 June 2019; pp. 6105–6114.
13. Radosavovic, I.; Kosaraju, R.P.; Girshick, R.; He, K.; Dollár, P. Designing network design spaces. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 10428–10436.
14. Szegedy, C.; Liu, W.; Jia, Y.; Sermanet, P.; Reed, S.; Anguelov, D.; Erhan, D.; Vanhoucke, V.; Rabinovich, A. Going deeper with convolutions. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 1–9.
15. Kang, E.; Han, Y.; Oh, I.S. Mushroom image recognition using convolutional neural network and transfer learning. *KIISE Trans. Comput. Pract.* **2018**, *24*, 53–57. [CrossRef]

16. Xiao, J.; Zhao, C.; Li, X.; Liu, Z.; Pang, B.; Yang, Y.; Wang, J. Research on mushroom image classification based on deep learning. *Softw. Eng.* **2020**, *23*, 21–26.
17. Shen, R.; Huang, Y.; Wen, X.; Zhang, L. Mushroom classification based on Xception and ResNet50 models. *J. Heihe Univ.* **2020**, *11*, 181–184.
18. Shuaichang, F.; Xiaomei, Y.; Jian, L. Toadstool image recognition based on deep residual network and transfer learning. *J. Transduct. Technol.* **2020**, *33*, 74–83.
19. Yuan, P.; Shen, C.; Xu, H. Fine-grained mushroom phenotype recognition based on transfer learning and bilinear CNN. *Trans. Chin. Soc. Agric. Mach.* **2021**, *52*, 151–158.
20. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, Ł.; Polosukhin, I. Attention is all you need. In Proceedings of the Advances in Neural Information Processing Systems, Long Beach, CA, USA, 4–9 December 2017, Volume 30.
21. Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S.; et al. An image is worth 16 × 16 words: Transformers for image recognition at scale. *arXiv* **2020**, arXiv:2010.11929.
22. Touvron, H.; Cord, M.; Douze, M.; Massa, F.; Sablayrolles, A.; Jégou, H. Training data-efficient image transformers & distillation through attention. In Proceedings of the International Conference on Machine Learning, Virtual, 18–24 July 2021; pp. 10347–10357.
23. Dong, X.; Bao, J.; Chen, D.; Zhang, W.; Yu, N.; Yuan, L.; Chen, D.; Guo, B. Cswin transformer: A general vision transformer backbone with cross-shaped windows. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022; pp. 12124–12134.
24. Yang, J.; Li, C.; Zhang, P.; Dai, X.; Xiao, B.; Yuan, L.; Gao, J. Focal self-attention for local-global interactions in vision transformers. *arXiv* **2021**, arXiv:2107.00641.
25. Xu, R.; Tu, Z.; Xiang, H.; Shao, W.; Zhou, B.; Ma, J. CoBEVT: Cooperative bird's eye view semantic segmentation with sparse transformers. *arXiv* **2022**, arXiv:2207.02202.
26. Yuan, L.; Chen, Y.; Wang, T.; Yu, W.; Shi, Y.; Jiang, Z.H.; Tay, F.E.; Feng, J.; Yan, S. Tokens-to-token vit: Training vision transformers from scratch on imagenet. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, QC, Canada, 10–17 October 2021; pp. 558–567.
27. Li, Y.; Wu, C.Y.; Fan, H.; Mangalam, K.; Xiong, B.; Malik, J.; Feichtenhofer, C. Improved multiscale vision transformers for classification and detection. *arXiv* **2021**, arXiv:2112.01526.
28. Wang, W.; Xie, E.; Li, X.; Fan, D.P.; Song, K.; Liang, D.; Lu, T.; Luo, P.; Shao, L. Pyramid vision transformer: A versatile backbone for dense prediction without convolutions. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, QC, Canada, 10–17 October 2021; pp. 568–578.
29. Xu, R.; Xiang, H.; Tu, Z.; Xia, X.; Yang, M.H.; Ma, J. V2X-ViT: Vehicle-to-everything cooperative perception with vision transformer. In Proceedings of the Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, 23–27 October 2022; pp. 107–124.
30. Bello, I.; Fedus, W.; Du, X.; Cubuk, E.D.; Srinivas, A.; Lin, T.Y.; Shlens, J.; Zoph, B. Revisiting resnets: Improved training and scaling strategies. *Adv. Neural Inf. Process. Syst.* **2021**, *34*, 22614–22627.
31. Touvron, H.; Cord, M.; Sablayrolles, A.; Synnaeve, G.; Jégou, H. Going deeper with image transformers. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, QC, Canada, 10–17 October 2021; pp. 32–42.
32. Zhou, D.; Kang, B.; Jin, X.; Yang, L.; Lian, X.; Jiang, Z.; Hou, Q.; Feng, J. Deepvit: Towards deeper vision transformer. *arXiv* **2021**, arXiv:2103.11886.
33. Wang, B. Automatic Mushroom Species Classification Model for Foodborne Disease Prevention Based on Vision Transformer. *J. Food Qual.* **2022**, *2022*, 1173102. [CrossRef]
34. Tan, M.; Le, Q. Efficientnetv2: Smaller models and faster training. In Proceedings of the International Conference on Machine Learning, Virtual, 18–24 July 2021; pp. 10096–10106.
35. Brock, A.; De, S.; Smith, S.L.; Simonyan, K. High-performance large-scale image recognition without normalization. In Proceedings of the International Conference on Machine Learning, Virtual, 18–24 July 2021; pp. 1059–1071.
36. Mehta, S.; Rastegari, M. Mobilevit: Light-weight, general-purpose, and mobile-friendly vision transformer. *arXiv* **2021**, arXiv:2110.02178.
37. Tu, Z.; Talebi, H.; Zhang, H.; Yang, F.; Milanfar, P.; Bovik, A.; Li, Y. Maxvit: Multi-axis vision transformer. In Proceedings of the Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, 23–27 October 2022; pp. 459–479.
38. Kiss, N.; Czúni, L. Mushroom image classification with CNNs: A case-study of different learning strategies. In Proceedings of the 2021 12th International Symposium on Image and Signal Processing and Analysis (ISPA), Zagreb, Croatia, 13–15 September 2021; pp. 165–170.
39. Zhuang, F.; Qi, Z.; Duan, K.; Xi, D.; Zhu, Y.; Zhu, H.; Xiong, H.; He, Q. A comprehensive survey on transfer learning. *Proc. IEEE* **2020**, *109*, 43–76. [CrossRef]
40. Liu, H.; Dai, Z.; So, D.; Le, Q.V. Pay attention to mlps. *Adv. Neural Inf. Process. Syst.* **2021**, *34*, 9204–9215.
41. Chu, X.; Tian, Z.; Zhang, B.; Wang, X.; Wei, X.; Xia, H.; Shen, C. Conditional positional encodings for vision transformers. *arXiv* **2021**, arXiv:2102.10882.

42. Liu, Z.; Mao, H.; Wu, C.Y.; Feichtenhofer, C.; Darrell, T.; Xie, S. A convnet for the 2020s. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022; pp. 11976–11986.
43. Selvaraju, R.R.; Cogswell, M.; Das, A.; Vedantam, R.; Parikh, D.; Batra, D. Grad-cam: Visual explanations from deep networks via gradient-based localization. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 618–626.

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