forests

Abstract: Timely detection of forest wildfires is of great significance to the early prevention and control of large-scale forest fires. Unmanned Aerial Vehicle(UAV) with cameras has the characteristics of wide monitoring range and strong flexibility, making it very suitable for early detection of forest fire. However, the visual angle/distance of UAV in the process of image sampling and the limited sample size of UAV labeled images limit the accuracy of forest fire recognition based on UAV images. This paper proposes a FT-ResNet50 model based on transfer learning. The model migrates the ResNet network trained on an ImageNet dataset and its initialization parameters into the target dataset of forest fire identification based on UAV images. Combined with the characteristics of the target data set, Adam and Mis functions are used to fine tune the three convolution blocks of ResNet, and focal loss function and network structure parameters are added to optimize the ResNet network, to extract more effectively deep semantic information from fire images. The experimental results show that compared with baseline models, FT-ResNet50 achieved better accuracy in forest fire identification. The recognition accuracy of the FT-ResNet50 model was 79.48%; 3.87% higher than ResNet50 and 6.22% higher than VGG16.

Keywords: forest fire recognition; transfer learning; sample augmentation; ResNet50

1. Introduction

Wild forest fires occur frequently all over the world. Forest fires usually have the characteristics of high risk and strong destructive potential, and pose a great harm to social and economic development, environmental protection and ecosystems. Different from other fires, forest fires present specific damage modes due to their environment. In the open environment and with sufficient oxygen, fires are more likely to occur and spread in forests, causing serious personal safety risks and economic losses. Early fire detection is the only effective way to reduce the harm of forest fires [1]. Therefore, research on forest fire identification and early warning has attracted extensive attention.

At present, forest fire detection is mainly realized through monitoring towers, aviation and satellite systems, optical sensors, digital cameras and wireless sensor networks [2,3]. However, forest fire detection methods based on monitoring towers largely depend on the experience of observers, and it is difficult for the monitoring range to cover a large area of wild forest. Satellite remote sensing is very effective for detecting large-scale forest fires, but it is limited by the difficulty in effectively identifying early regional fires [4,5]. Fire detection systems based on sensor networks have good identification performance in indoor spaces, but are difficult to install and maintain in wild forest areas due to high hardware costs [6,7]. In addition, due to the limitations of sensor materials, interference from the environment may lead to false positives. At the same time, wireless sensor networks are unable to provide important visual information to help firefighters track the fire scene. In recent years, with the development of machine vision technology, researchers have proposed various fire detection models based on image processing [8,9]. However, image processing methods based on fixed cameras are limited by the field of view, which leads to poor monitoring ability for large scenes. In addition, undulating terrain in forests may block the scenes of some fires. Different from the above forest fire detection methods, unmanned aerial vehicles (UAVs) equipped with cameras can solve the problems of fixed position cameras, can cover a larger monitoring range, and are not limited by the installation angle [10,11]. Therefore, UAV images are especially suitable for early recognition of forest fire.

Many countries have carried out relevant research and practical activities for forest fire monitoring and identification based on UAVs [12]. Fire identification technology based on UAV images is usually developed based on the color, motion and geometric

features of the images [13]. Jiao et al. proposed a forest fire detection algorithm using YOLOv3 to extract color and shape features from aerial images taken by unmanned aircraft [14]. Anh et al. proposed a method based on RGB color space to distinguish fire pixels and background [15]. Yuan et al. used median filter, color space conversion, Otsu threshold segmentation, morphological operations and blob counter to detect and track potential fires in sequence [16].

With the successful application of deep learning technology in the fields of intelligent transportation systems [17], indoor target positioning [18], and intelligent agriculture [19], researchers have also introduced deep learning technology into the field of forest fire detection to improve the accuracy of forest fire identification by extracting deep semantic features from images. Hu et al. proposed the MVMNet model to improve the accuracy and effectiveness of forest fire smoke target detection [20]. Guan et al. proposed a FireColorNet model based on color attention to extract color feature information from forest fire images [21]. Li et al. proposed an adversarial fusion network to extract abstract features for forest fire smoke detection [22]. Fan et al. proposed YOLOv4-Light, a lightweight network structure for forest fire detection. YOLOv4-Light uses MobileNet to replace the backbone feature extraction network of YOLOv4, and deep separable convolution to replace the standard convolution of PANet [23]. Federico et al. developed a deep learning model for forest fire detection, obtained from transfer-learning of pre-trained RetinaNet, and established a Faster R-CNN model for object detection [24]. However, the images captured by UAVs were overhead images. In the early detection of forest fires, the forest fire target is very small, and the color and shape characteristics are not obvious. Therefore, the above fire detection models based on color and shape features cannot be directly applied to UAV images. This brings great challenges for research into forest fire early recognition based on UAV images. In addition the lack of sufficient labeled UAV fire image samples directly affects the accuracy of forest fire recognition based on UAV images. At the same time, the lack of sufficient fire annotation image samples makes it difficult effectively to introduce deep learning methods into UAV image recognition, because these methods require large quantities of high-quality annotation data to obtain satisfactory recognition results.

In order to extract deeper abstract features from images, it is necessary to construct a deep-level network model, and the training of a deep neural network is a time-consuming and complex process. In addition, the training of a deep model needs a large number of labeled samples. This has become the bottleneck in the task of forest fire identification based on UAV images, and the emergence of transfer learning technology provides an opportunity to solve this problem. Transfer learning [25,26] refers to transferring the trained model to a new task, and realizing the modeling of the new task by fine-tuning the model parameters. When there are insufficient labeled samples, transfer learning can solve the problem of overfitting training caused by too few labeled samples.

In this paper, the idea of transfer learning is introduced into the research of forest fire recognition based on UAV images, and a new forest fire recognition model is proposed: FT-ResNet50, based on transfer learning. The Ft-ResNet50 model is based on the transfer learning method, and ResNet50 pre-trained on the ImageNet dataset is used as the backbone framework for forest fire recognition. The pre-trained weights are used as initialization parameters for the backbone network, and the original network is improved by optimizing network structure parameters. Finally, the optimized network is applied to the data-enhanced UAV forest fire dataset realize the effective identification of forest fire. The main contributions of this paper are:

(1) The FT-ResNet50 model adopted transfer learning to solve the problem of insufficient labeled samples of UAV forest fire images. This model can also realize high-performance forest fire recognition when UAV labeled samples are limited in size and uneven in sample distribution.

(2) The FT-ResNet50 selected ResNet50 as the basic network to realize transfer learning through experimental results. By fixing the shallow layers of ResNet50 and fine tuning its deep layers, we obtained the optimal configuration of ResNet50 suitable for the target dataset. The FT-ResNet50 model could successfully extract deep semantic features from UAV images, thus improving the accuracy of the model for forest fire recognition.
(3) The FT-ResNet50 model combined the mixup-based sample enhancement method with the traditional sample enhancement method to expand the sample size of UAV images, so as to enhance the generalization ability of the model.

The structure of this paper is organized as follows. In Section 2, the dataset used in the experiments is presented, and the structure of the FT-ResNet50 model is discussed in detail. Section 3 introduces the configuration of the experiment, and experimentally verifies the influence on forest fire identification of configuration parameters such as network depth, loss function, activation function, and optimizer, to explain the framework of the FT-ResNet50 model. In Section 4, the experimental results are discussed in depth and analyzed; Section 5 summarizes the full work.

2. Materials and Methods

2.1. Dataset

The FLAME (Fire Luminosity Airborne-based Machine learning Evaluation) dataset is a dataset of fire images collected by UAV in an Arizona pine forest [27]. The dataset used different UAVs and cameras to collect image samples of forest fires. Table 1 describes the technical specifications of UAVs and cameras used in the FLAME dataset, and the resolution of the collected samples. The dataset includes video recording and heat maps taken by infrared camera. Each frame of the video is labeled as an image. In this paper, 31501, 7874 and 8617 image samples were extracted from the FLAME data set as the training set, verification set and test set of this experiment, respectively. Figure 1 shows some examples of typical forest fire images in the FLAME dataset.

Table 1. Technical specifications of the UAVs and cameras used in the FLAME dataset and the resolution of the samples acquired.

UAVs	Cameras	Resolution
Phantom 3 Professional/ Matrice 200	FLIR camera	640 × 512
	Zenmuse camera	1280 × 720
	Phantom camera	3480 × 2160

Figure 1. Some image samples of forest fires in the FLAME dataset.

2.2. Mixup

In order to give the forest fire recognition model better generalization ability, this paper presents an expansion strategy for the forest fire image training samples. By increasing the number of training samples, the distribution of training samples can be improved and the robustness of the model to noise can be improved.

Zhang et al. [28] proposed a sample enhancement method based on mixup. This method is a sample expansion algorithm for computer vision, which can expand the size of a dataset by mixing different types of images. Two image samples are randomly selected from the training dataset, and the pixel values and labels of the two image samples are weighted according to a certain weight. Specifically, the mixup builds virtual training samples in the following ways:

$$\tilde{x} = \lambda x_i + (1-\lambda)x_j \tag{1}$$

$$\tilde{y} = \lambda y_i + (1-\lambda)y_j \tag{2}$$

$$Beta(\alpha,\beta)\ \ \alpha\in(0,\ \infty)\ ,\ \ \beta\in(0,\ \infty) \tag{3}$$

Where (x_i, y_j) and (x_i, y_j) are the two random examples extracted from training sample data, and $\lambda \in [0,1]$. λ follows a Beta distribution, namely $\lambda \sim Beta(\alpha,\alpha)$. This mixup-based data enhancement method has the advantages of processing decision boundary blurring, providing smoother predictions, and enhancing the prediction ability of the model beyond the scope of training dataset. Figure 2 shows the process of mixup-based image sample augmentation. The experiment shows that when λ = 0.5, $\alpha = \beta$, the best data-fusion effect is achieved.

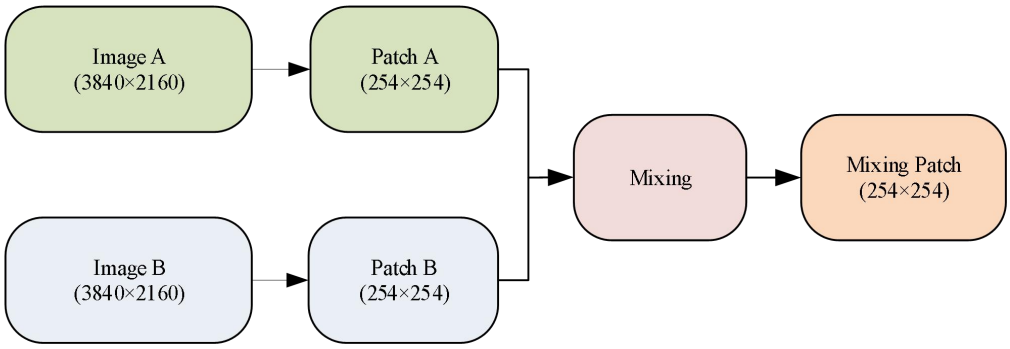


Figure 2. Image sample augmentation process based on mixup.

2.3. Residual Network (ResNet-50)

Following the success of VGGNet architectures [29], researchers believe that deeper models outperform shallower models. However, as the number of model layers increases, the complexity and training difficulty of the model also increases, and the accuracy decreases. In 2016, Kaiming He and colleagues at Microsoft Research solved the problem of gradient disappearance and gradient explosion by building ResNet, making feasible deeper network training. They introduced a new learning framework to simplify the training of deeper networks [30], and called the framework residual learning; accordingly, the model using this framework is called residual network (ResNet). ResNet allows original input information to be directly connected to subsequent neurons, and takes as its goal minimization of the difference (residual) between input and output. Specifically, the original input to the network is set to x and the final desired output is set to $H(x)$. When the original input x is passed directly to

the tail of the network as the initial result, the objective to be learned in this case becomes $F(x) = H(x) - x$. Figure 3 illustrates the principle of residual learning in ResNet.

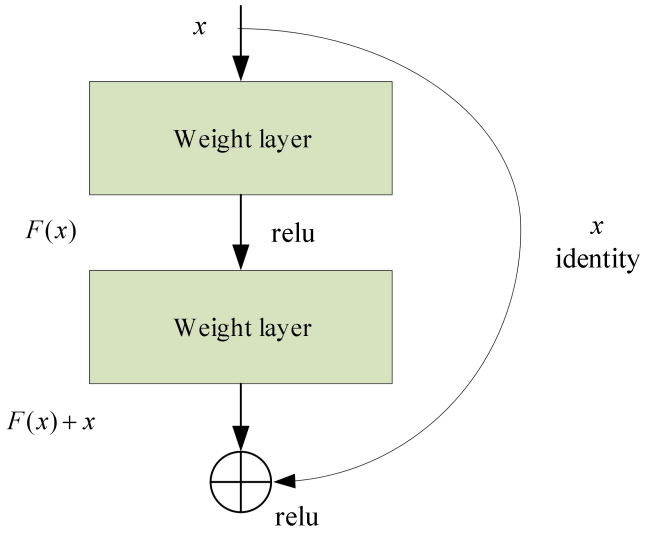


Figure 3. Residual learning module for ResNet.

This paper is devoted to extracting deeper semantic information from forest fire images, beyond color and structural features, so the ResNet-50 network was selected as the backbone network of our model. Table 2 lists the architecture of ResNet-50. ResNet-50 contains 49 convolution layers, one of which is 3 × 3, an average pool layer, and a fully connected layer. The classical ResNet-50 model involves 25.56 million parameters, of which the rectification nonlinearity (ReLU) activation function and batch normalization (BN) function are applied to the back of all convolution layers in the "Bottle-neck" block, and the softmax function is applied to the full connection layer.

Table 2. Network configuration for ResNet-50.

Layer Name	Output Size	50-Layer
Conv 1	112 × 112	7 × 7, 64, stride 2
Conv 2_x	56 × 56	3 × 3 max pool, stride 2 $\begin{bmatrix} 1 \times 1,64 \\ 3 \times 3,64 \\ 1 \times 1,256 \end{bmatrix} \times 3$
Conv 3_x	28 × 28	$\begin{bmatrix} 1 \times 1,128 \\ 3 \times 3,128 \\ 1 \times 1,512 \end{bmatrix} \times 4$
Conv 4_x	14 × 14	$\begin{bmatrix} 1 \times 1,256 \\ 3 \times 3,256 \\ 1 \times 1,1024 \end{bmatrix} \times 6$
Conv 5_x	7 × 7	$\begin{bmatrix} 1 \times 1,512 \\ 3 \times 3,512 \\ 1 \times 1,2048 \end{bmatrix} \times 3$
	1 × 1	Average pool, 1000-d fc, softmax
FLOPs		3.8×10^9

2.4. Transfer Learning

The idea of transfer learning was introduced to solve the problem of limited sample size of UAV forest fire images. In this study, the ResNet50 network trained on ImageNet

dataset [31] was migrated to the experimental dataset of UAV forest fires. The ImageNet dataset contains about 1.2 million images in 1000 categories. Using the network model pre-trained on such a large dataset, it can be effectively migrated to classification tasks of various images [32]. The ResNet50 network was trained on the Imagenet dataset, taken as the preliminary training model, and the optimal configuration of the ResNet50 network was realized by fixing the convolution block of shallow feature extraction, fine-tuning the convolution block of deep feature extraction, and adjusting the Mish and Adam parameters, to complete feature extraction and recognition based on UAV forest fire images.

2.5. Adam Optimizer

In this study, an Adam optimizer was used to accelerate the convergence of the FT-ResNet50 model. Adam is a first-order gradient-based stochastic objective function optimization algorithm [33]. Adam combines the advantages of the AdaGrad [34] and RMSProp [35] algorithms; the former is used for sparse gradient problems, and the latter is used for nonlinear and unfixed optimization objective problems. Adam has the advantages of easy implementation, high computing efficiency and low memory requirements [36]. Its gradient diagonal scaling is invariant, so it is suitable for solving problems with large-scale data or parameters. For different parameters, Adam can adaptively adjust the learning rate and iteratively update the weights of the neural network according to the training data [37,38]. The calculation process and pseudocode of the Adam algorithm are shown in Algorithm 1.

Algorithm 1. The calculation process and pseudocode of Adam algorithm.

g_t^2 indicates the elementwise square ($g_t \odot g_t$). Good default settings for the tested machine learning problems are α = 0.001, $\beta 1$ = 0.9, $\beta 2$ = 0.999, $\epsilon = 10^{-8}$. All operations on vectors are element-wise. With β_1^t and β_2^t we denote $\beta 1$ and $\beta 2$ to the power t:

Require: α: Stepsize
Require: $\beta_1, \beta_2 \in [0, 1)$: Exponential decay rates for the moment estimates
Require: f(θ): Stochastic objective function with parameters θ
Require: θ_0: Initial parameter vector
$m_0 \leftarrow 0$ (Initialize 1st moment vector)
$v_0 \leftarrow 0$ (Initialize 2nd moment vector)
t ← 0 (Initialize timestep)
while θ_t not converged **do**
t ← t + 1
$g_t \leftarrow \nabla_\theta f_t(\theta_{t-1})$ (Get gradient w.r.t stochastic objective at timestep t)
$m_t \leftarrow \beta_1 \cdot m_{t-1} + (1 - \beta_1) \cdot g_t$ (Update biased first moment estimate)
$v_t \leftarrow \beta_2 \cdot v_{t-1} + (1 - \beta_2) \cdot g_t^2$ (Update biased second raw moment estimate)
$mb_t \leftarrow m_t / (1 - \beta_1^t)$ (Compute bias-corrected first moment estimate)
$vb_t \leftarrow v_t / (1 - \beta_2^t)$ (Compute bias-corrected second raw moment estimate)
$\theta_t \leftarrow \theta_{t-1} - a \cdot mb_t / (\sqrt{v}\, b_t + \varepsilon)$ (Update parameters)
end while
return θ_t (Resulting parameters)

2.6. Focal Loss

Focal Loss function [39] is mainly used to solve problems such as unbalanced sample number and sample difficulty. When training the FT-ResNet50 model, Focal Loss was used as a loss function to update ω and b. The Focal Loss function is defined as follows:

$$FL(p_t) = -\alpha_t (1 - p_t)^\gamma \log(p_t) \quad (4)$$

p_t reflects the proximity to ground truth. The larger p_t is, the closer it is to the ground truth, i.e., the more accurate the classification. γ is the adjustable factor. Focal Loss's modulation factor is $(1 - p_t)^\gamma$; for the accurately classified sample $p_t \to 1$, modulating factor approaches 0; for the inaccurately classified sample $1 - p_t \to 1$, modulating factor approaches 1. Compared with traditional cross entropy loss, Focal Loss does not change for samples with inaccurate classification, and for samples with accurate classification, loss decreases. On the whole, it is equivalent to increasing the weight of the inaccurately classified samples in the loss function.

p_t also reflects the difficulty of classification. The larger p_t, the higher the confidence of classification, and the easier it is to divide the sample. The smaller p_t is, the lower the confidence of classification, and the more difficult it is to distinguish the sample. Therefore, Focal Loss increases the weight of difficult samples in the loss function, making the loss function tend towards the difficult samples, which helps to improve the accuracy of difficult samples and improve the learning ability of the network for the current task.

2.7. Mish

Mish function [40] is a novel self-regularized non-monotonic activation function. Its shape and properties are similar to those of Swish. It plays an important role in the

performance and training dynamics of neural networks. The Mish activation function can be expressed as follows:

$$Mish(x) = x \tanh(\log(1 + e^x)) \quad (5)$$

Compared with the ReLU function, which is the common activation function in the neural network, Mish is differentiable anywhere in its domain, so there is no hard turning point at zero. Figure 4 shows the curve of the Mish function.

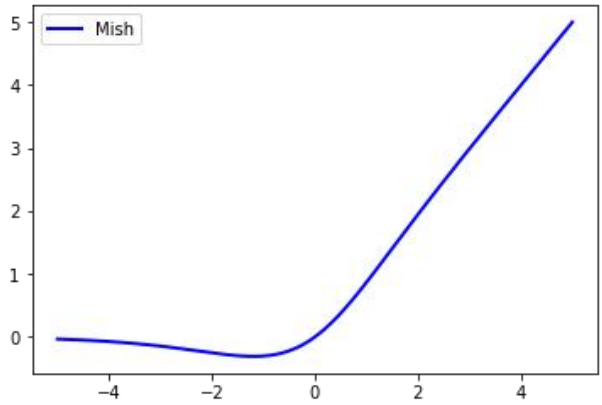


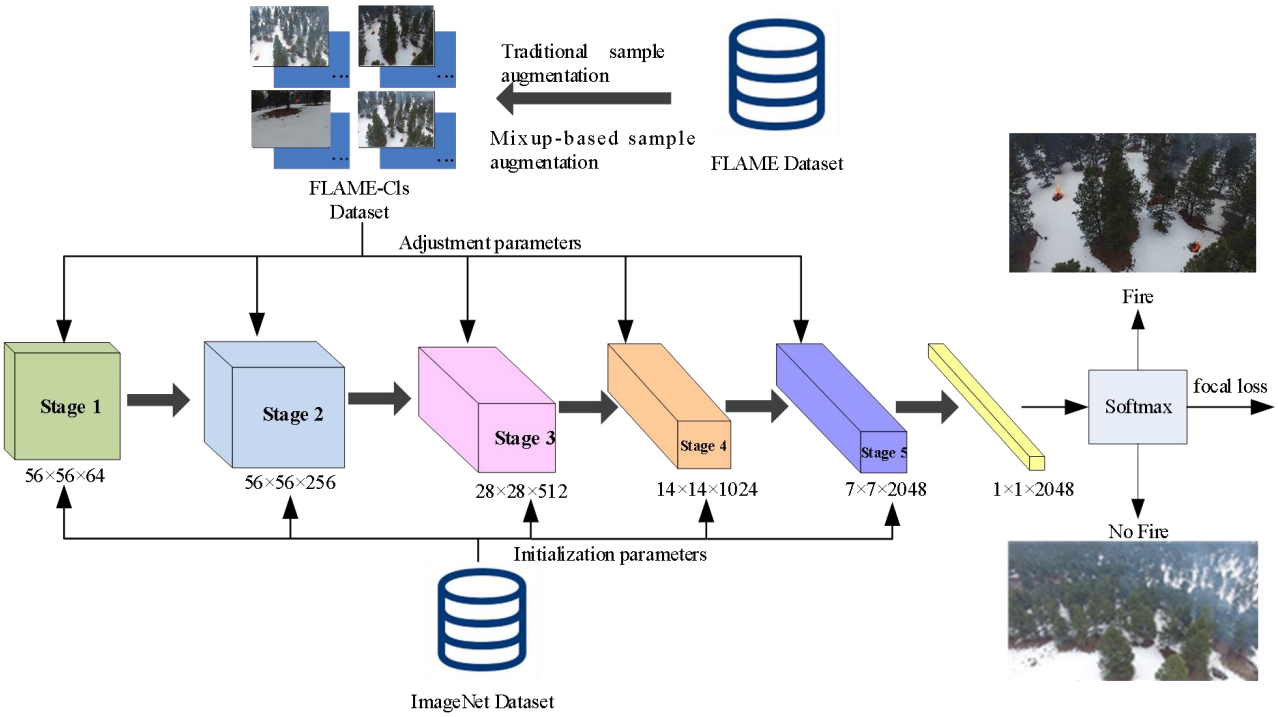
Figure 4. Curve of Mish function.

2.8. The Proposed Forest Fire Identification Model—FT-ResNet50

This section introduces the FT-ResNet50 model in detail. Figure 5 shows the architecture of the FT-ResNet50 model.

FLAME-Cls is the extended data set after sample enhancement. The FT-ResNet50 model uses five-level residual blocks for feature extraction. The first two residual blocks are mainly used to extract the edge, texture and color features of the image. Because the extraction process of these features is highly universal for all types of images, the structure of the first two-stage residual block is the same as that of ResNet50 in the FT-ResNet50 model. The next three-level residual block mainly extracts the abstract semantic features of the image, which is the key to improving the accuracy of forest fire recognition. The FT-ResNet50 model adjusts the last three residual blocks of the ResNet50 network, and adds the Adam random gradient descent algorithm to residual blocks 3, 4 and 5 to avoid training falling into local optimization, and to ensure that the model can obtain more accurate recognition results. The feature map output from the last convolution layer of the FT-ResNet50 model is converted into a 2048-dimensional vector through the global average pool, and the forest fire identification results are output in the form of probability through the SoftMax function.

Meanwhile, in the FT-ResNet50 model, the original activation function ReLu was replaced by the Mish function to improve the gradient vanishing problem in model training. In addition, the Focal Loss s was employed to replace the traditional binary cross-entropy loss. Focal Loss pays more attention to the training of difficult samples, which is more helpful for improving the learning ability of the model.



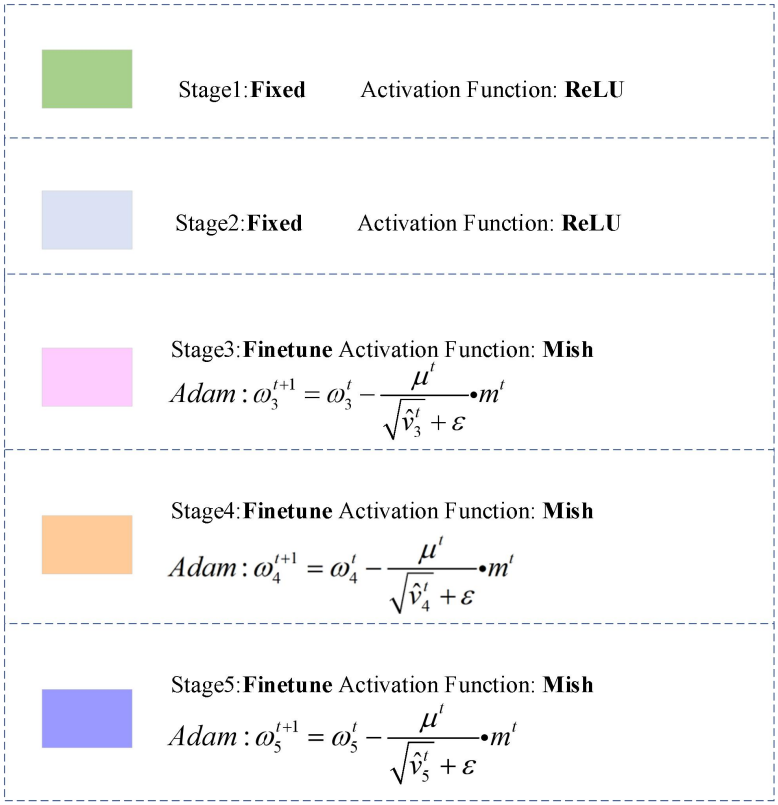


Figure 5. The architecture of the FT-ResNet50 model.

3. Results

3.1. Experiment Setup

3.1.1. Experimental Condition Configuration

Table 3 lists the experimental conditions. In order to verify the performance of the FT-ResNet50 model based on the enhanced FLAME-Cls dataset, this study compared the

recognition performance of the FT-ResNet50 model with VGG, Inception, and ResNet. Table 4 shows the setting of super parameters of the FT-ResNet50 model.

Table 3. Experimental condition configuration.

Experimental Enviroment	Details
Programming language	Python 3.8
Operating system	Ubuntu 18.04
Deep learning framework	PyTorch 1.7.0
GPU type	GeForce RTX 2080Ti
CPU type	Intel(R) Xeon(R) Silver 4110

Table 4. Hyperparameter settings.

Hyperparameters	Values
Batch size	32
Training epoch	40
Initial learning rate	0.001
Optimization algorithm	Adam
Activation function	Mish

3.1.2. Evaluation indicators

In order to evaluate comprehensively the effect of the forest fire identification method proposed in this paper, we used accuracy (Acc), precision (Pre), recall (Rec), specificity (Spe), and F1 score as evaluation indicators shown in Equations (6)–(10). True negative (TN), called the true negative rate, indicates the number of samples among the negative samples that are actually predicted to be negative. False positive (FP), called the false positive rate, indicates the number of samples among the negative samples that are actually predicted to be positive. False negative (FN), called the false negative rate, indicates the number of samples among the positive samples that are actually predicted to be negative. True positive (TP), called the true positive rate, indicates the number of samples among the positive samples that are actually predicted to be positive.

$$Acc = \frac{TP + TN}{TP + FN + FP + TN} \quad (6)$$

$$Pre = \frac{TP}{TP + FP} \quad (7)$$

$$Rec = \frac{TP}{TP + FN} \quad (8)$$

$$Spe = \frac{TN}{TN + FP} \quad (9)$$

$$F_1 = \frac{2 * Pre * Rec}{Pre + Rec} \quad (10)$$

3.2. Experimental Results

3.2.1. Sample Augmentation

In this study, the traditional sample augmentation method was combined with the mixup-based sample augmentation method to expand the samples of the FLAME dataset, and a new forest fire dataset, FLAME-Cls, was obtained after the sample augmentation. Figures 6 and 7 show the expansion effects of the traditional sample augmentation method and the mixup-based sample augmentation method, respectively.

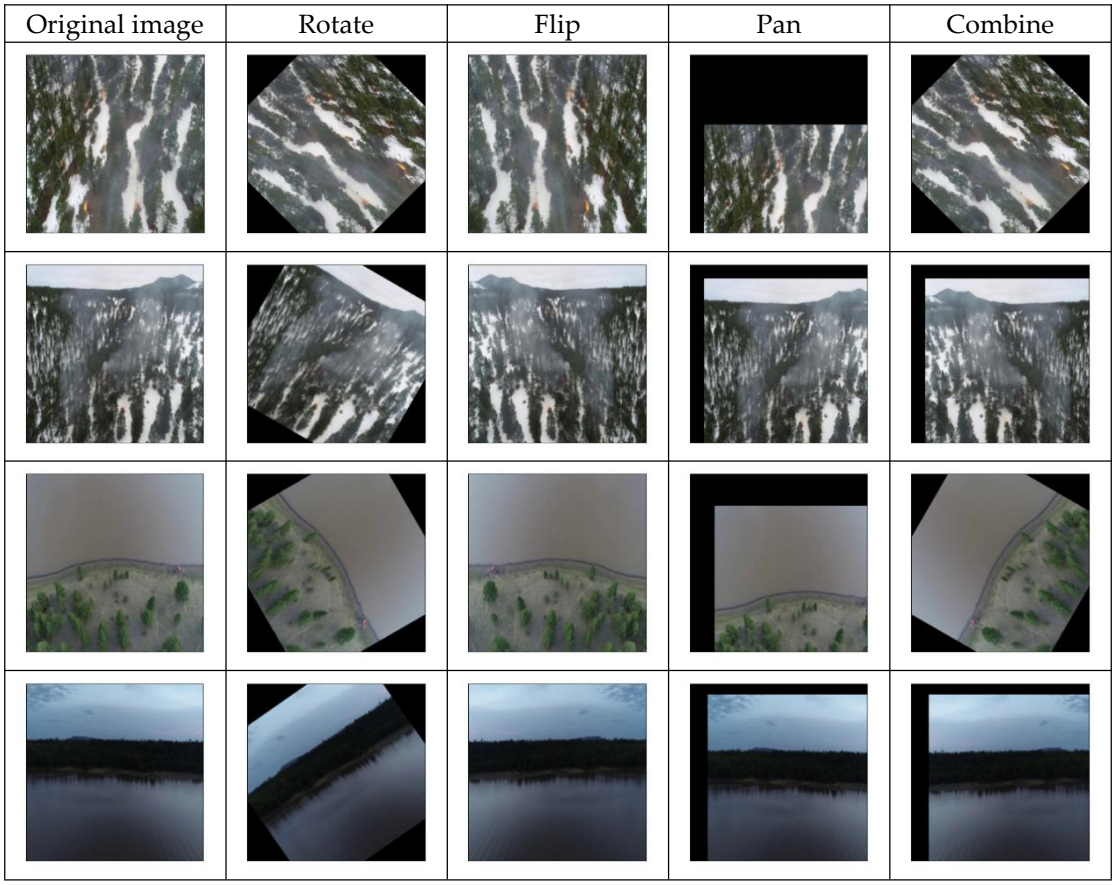


Figure 6. Traditional sample augmentation.

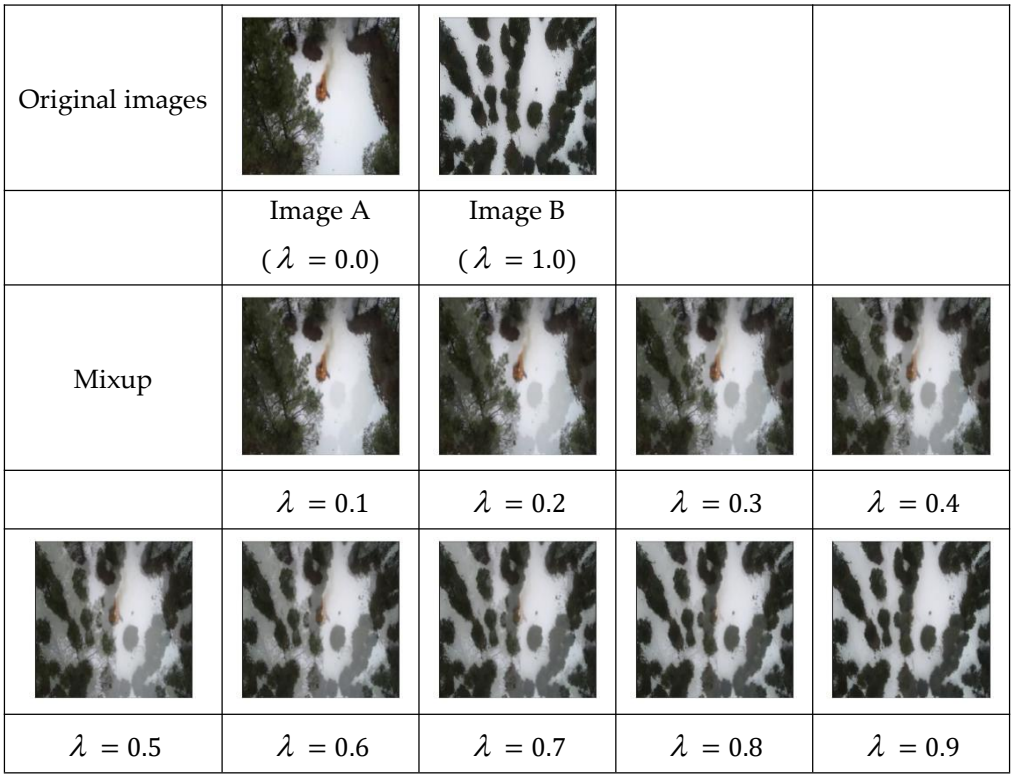


Figure 7. Mixup-based sample augmentation.

Using ResNet50 as the forest fire identification model (the parameters in each convolutional block determined by the ImageNet dataset), the effects of different sample augmentation methods were verified, and the results are shown in Table 5. As an online augmentation strategy (as the training process proceeds), mixup does not change the number of training samples in each round. The traditional augmentation scheme (offline expansion, supplementing the number of samples in the basic training set) can improve identification accuracy to a certain extent (73.66% to 75.18%) but also brings additional training costs. We finally adopted a combination of two augmentation strategies and achieved a level of performance improvement, namely 77.47% recognition accuracy.

Table 5. Influence of sample augmentation strategy on forest fire identification accuracy.

Augmentation Strategy	Number of Samples	Accuracy (%)
Original dataset	31,501	73.66
Traditional sample augmentation	50,000	75.18
Mixup-based sample augmentation	50,000	76.24
Proposed method	50,000	77.47

3.2.2. Forest Fire Identification Results

Table 6 shows the respective recognition accuracy and loss for the proposed method on the training set, validation set and test set. It can be seen that it achieved relatively good results with both the training set and the validation set, while the performance for the test set was relatively lower, reflecting a large domain offset between the test set data and the training and validation data, improving the generalization requirements of the model to a certain extent. Domain shift can be understood as the difference in data distribution between two sample sets. Generalization refers to how well a model has been tested on different distributions. For example, there is a large difference between training samples and test samples, so a robust model is needed to generalize samples that have not been seen before.

Table 6. Identification accuracy and loss for different datasets.

Dataset	Loss Function Value	Accuracy (%)
Training set	0.0612	97.14
Validation set	0.0398	97.71
Testing set	0.6823	79.48

The convergence curves of the loss function and the recognition accuracy with the number of iterations are shown in Figures 8 and 9. It can be seen that the performance of the proposed network in the validation set also improved, reflecting its relatively reliable generalization performance.

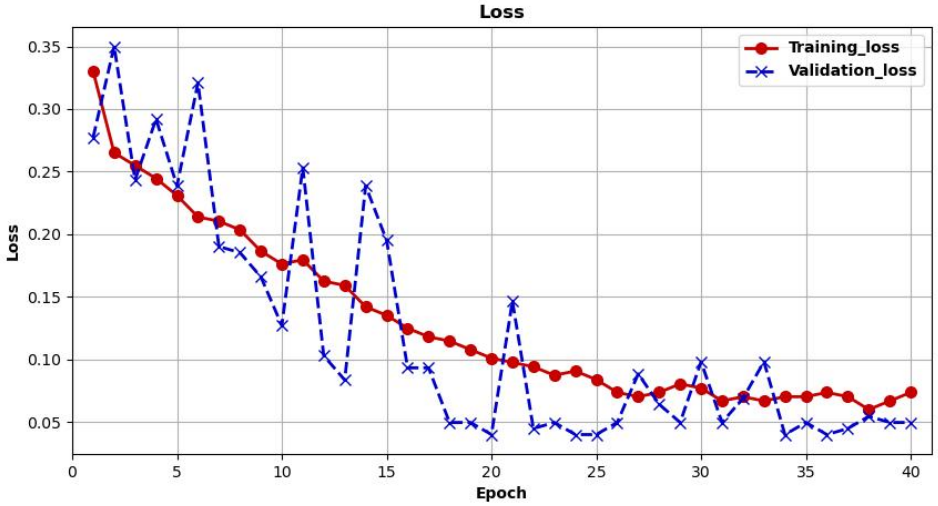


Figure 8. Convergence curves of the loss function.

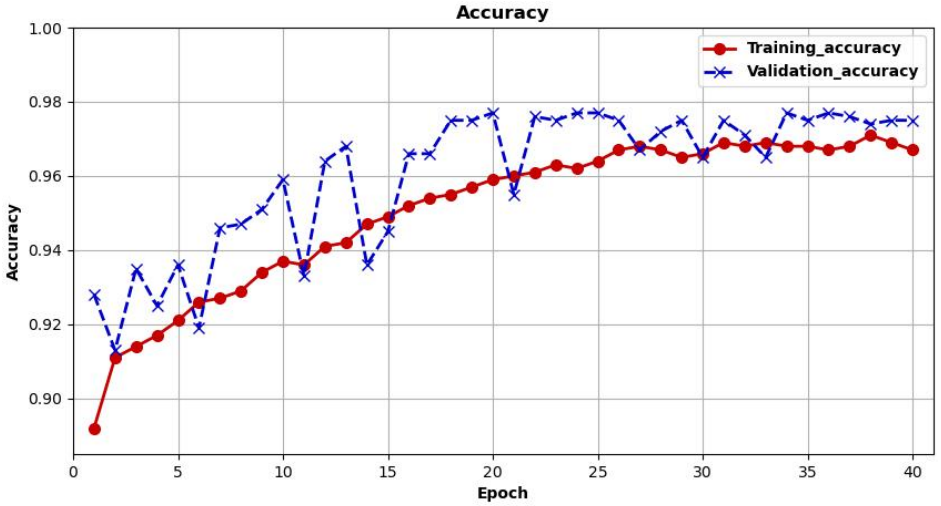


Figure 9. Identification accuracy with the number of iterations.

3.2.3. Influence of Different Backbone Networks on Recognition Accuracy

This research used VGG, Inception, and ResNet50 as the backbone network to test forest fire recognition rates. As shown in Table 7, compared with the other two structures, choosing ResNet50 with residual structure as the backbone network can to a certain extent alleviate the problem of gradient disappearance, which is beneficial for model training. ResNet50 replaces the fully connected layer in VGG with global average pooling in the final output, which greatly reduces the network parameters and the risk of overfitting. Figure 10 shows the comparison of confusion matrices for three different backbone network structures. It can be seen that ResNet50 achieved more accurate predictions than other methods.

Table 7. Recognition accuracy and loss for different datasets.

Network	Loss Function	Accuracy (%)	Parameter Quantity
VGG	BCE loss	72.12	138 M
	Focal loss	73.26	
Inception	BCE loss	74.87	23.2 M
	Focal loss	76.64	
ResNet	BCE loss	75.91	25.5 M

Focal loss	77.47

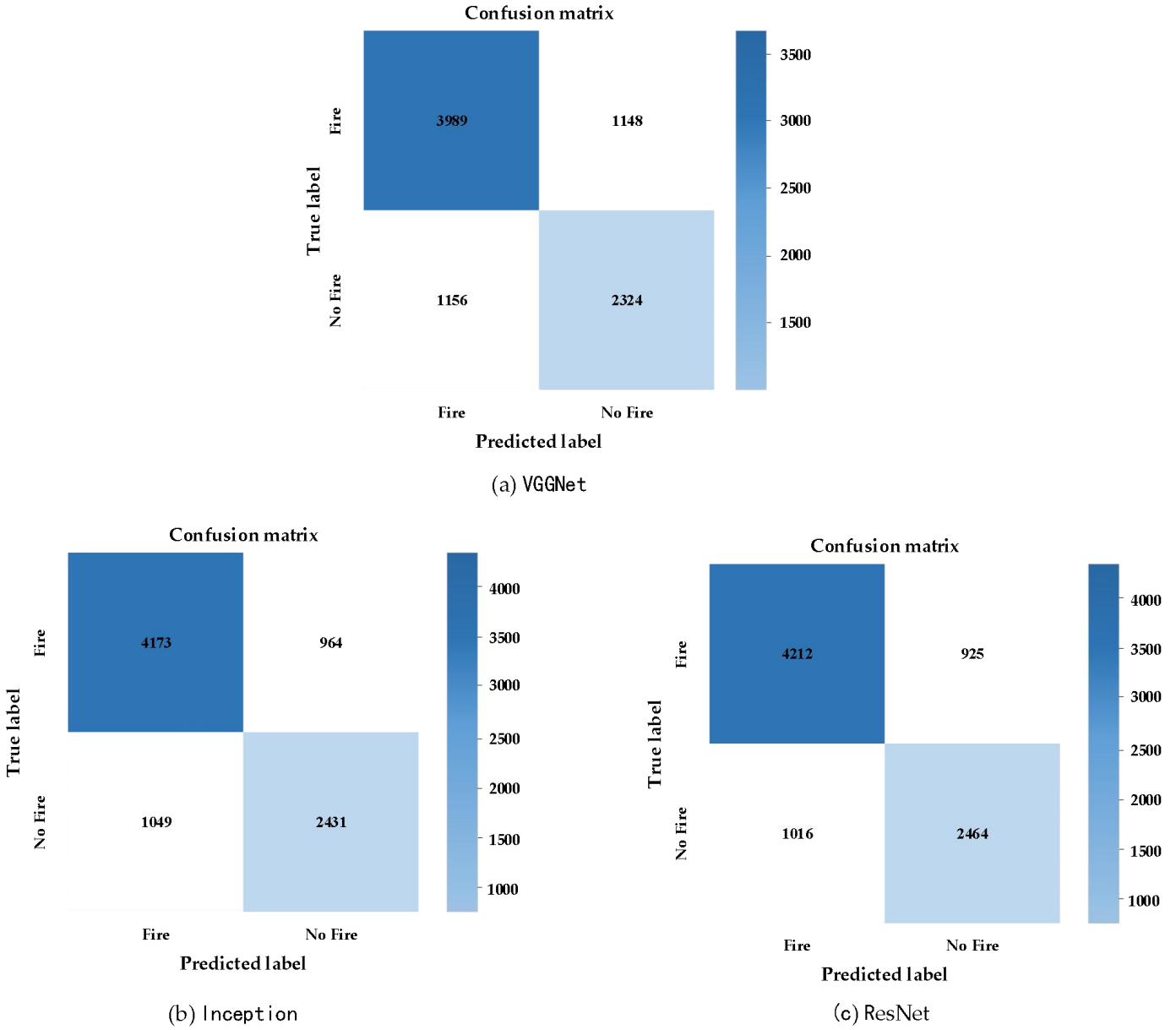


Figure 10. Comparison of confusion matrix of different backbone networks.

3.2.4. Influence of ResNet Network Depth on Identification Accuracy

This research explored the impact of network depth on model performance from the aspects of recognition accuracy and inference time. The experimental results are shown in Table 8. The backbone network used in this experiment was ResNet, and four different configuration structures were tested here, which are respectively represented as ResNet18, ResNet34, ResNet50, and ResNet101. Among these, ResNet18 and ResNet34 use the BasicBlock model structure, and the latter two use another model structure called Bottleneck. For deeper convolutional networks, the Bottleneck structure can reduce parameters to a certain extent and can prevent overfitting. Observing the data, it can be concluded that as the number of network layers deepens, the inference time also increases, that is, the efficiency of image recognition becomes lower. Although ResNet101 achieved the highest recognition accuracy, the inference speed was 1/3 slower than the second-ranked ReNet50. In summary, it is important to choose the most

appropriate network depth for a specific task. In this study, after balancing model complexity, training cost, test accuracy and other factors, ResNet50 was finally selected as the backbone network for feature extraction.

Table 8. Effects of different ResNet network depths on forest fire recognition performance.

Network	Accuracy (%)	Precision	Sensitivity	Specificity	F1 score	Inference Speed (FPS)
ResNet18	75.24	79.21	79.27	69.28	79.24	22.7
ResNet34	76.13	79.78	80.32	69.94	80.49	19.6
ResNet50	77.47	80.57	81.99	70.80	81.27	18.1
ResNet101	77.62	80.79	82.13	71.18	81.45	12.3

To further explore the impact of training methods on recognition accuracy, this study used different activation functions and optimizers for the testing, and the results are shown in Table 9. The experimental results show that the best recognition effect was provided by the combination of Mish and Adam. A schematic diagram of activation functions and the convergence curves for different optimizers are shown in Figures 11 and 12, respectively.

Table 9. Effects of different activation functions and optimizers on test accuracy.

Types	Schemes	Accuracy(%)	Rank
Activation functions	tanh	73.22	5
	Sigmoid	73.01	6
	ReLU	75.36	4
	LReLU	75.98	3
	Mish	76.82	1
	Swish	76.23	2
Optimizers	SGD	75.12	5
	Momentum	76.54	3
	RMSprop	76.69	2
	Adam	77.28	1
	Adagrad	75.31	4

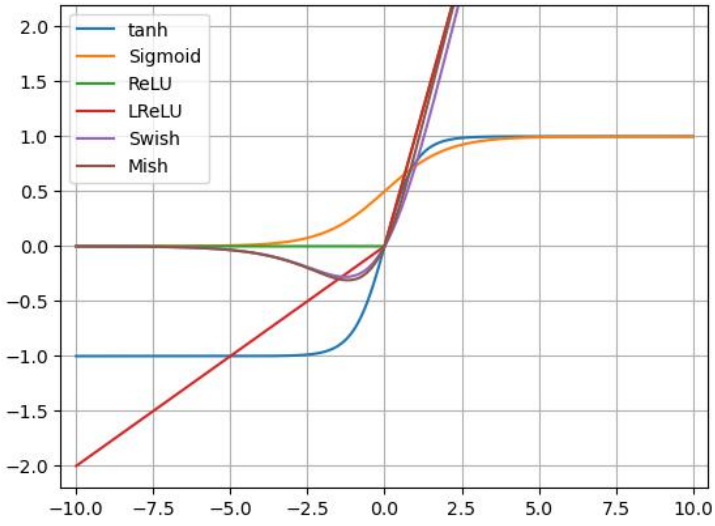


Figure 11. Schematic diagram of activation functions.

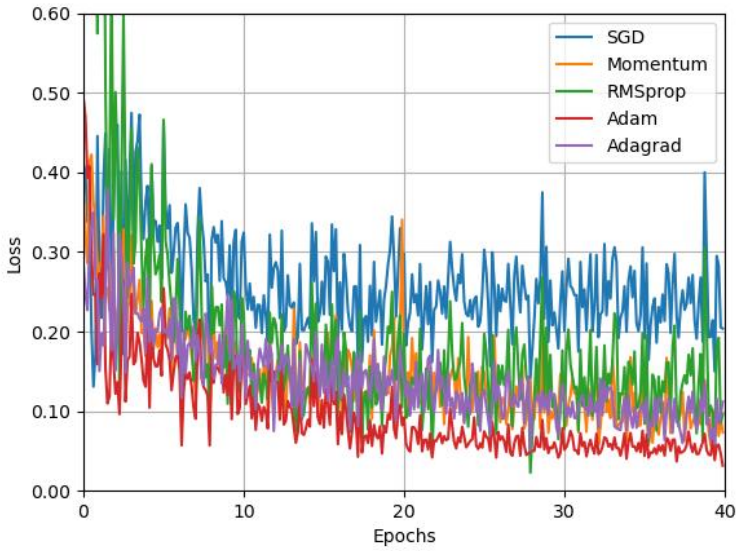


Figure 12. Convergence curves for different optimizers.

3.2.5. The Effect of Transfer Learning Strategy on Identification Accuracy

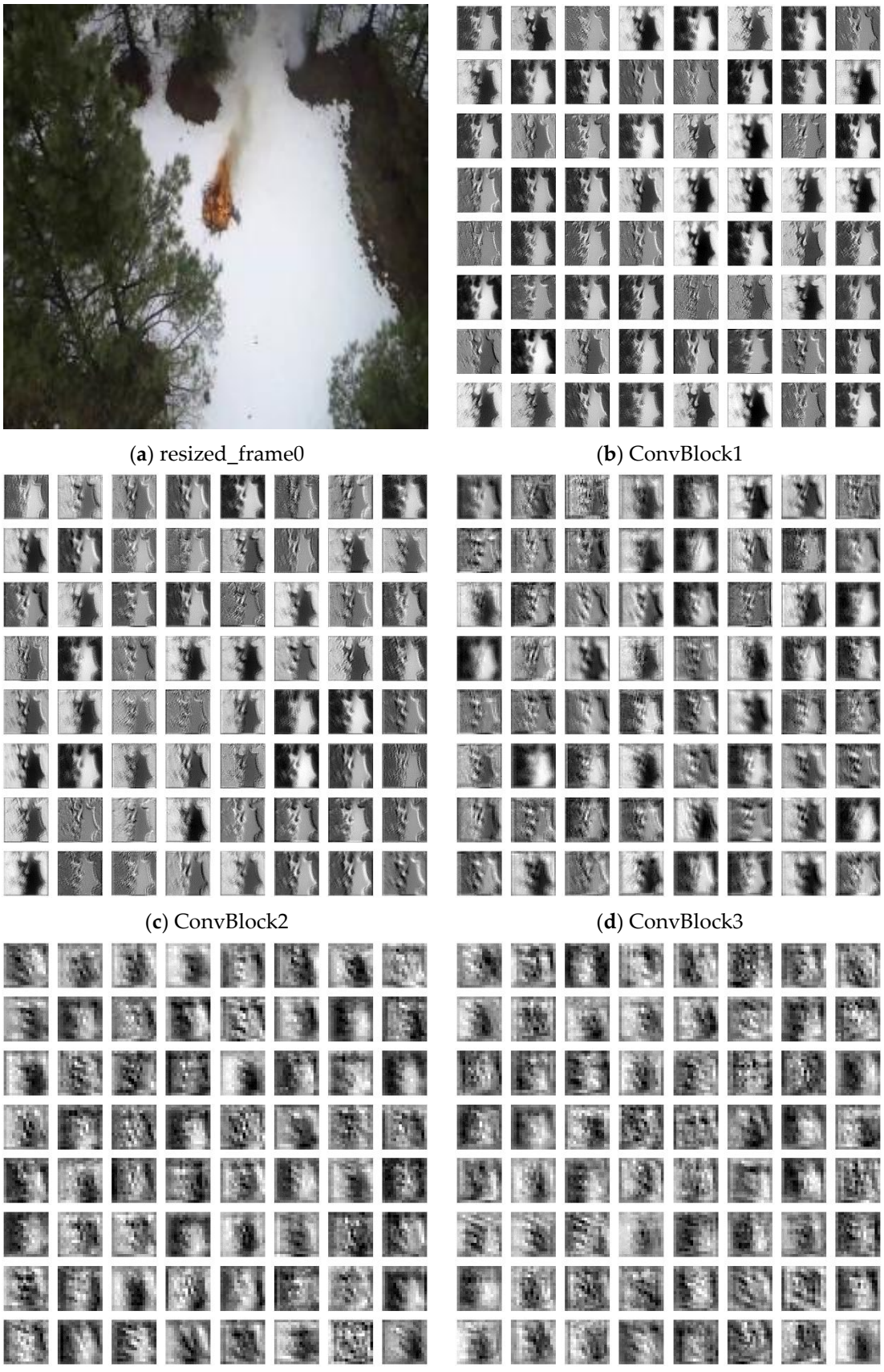
This paper explores the impact of transfer learning strategies on model testing accuracy under different network architectures. ResNet50 and VGG16 networks were used as backbone networks to extract features, both composed of five convolution blocks, denoted as ConvBlock1–ConvBlock5. The transfer learning strategy used the weight parameters of the original network model trained on the ImageNet dataset as the initialization parameters for the corresponding part of the FT-ResNet50 model proposed in this paper, and selected fixed and fine-tuned parameters on this basis to achieve the effect of accelerating network convergence. To this end, the authors designed six transfer learning schemes for each network model, as shown in Table 10.

In order to explore the method and process of extracting the features of a given input image by the model proposed in this paper, the features captured by the model backbone network ResNet50 were visualized as shown in Figure 13. It can be seen that ResNet50 can extract richer features. The combination of low-level edge information and high-level semantic information more specifically guides the process of feature acquisition in the target-related regions, thereby improving the accuracy of target classification.

Table 10. The impact of different transfer learning strategies on test accuracy.

Types	Schemes	Fixed	Fine-tune	Accuracy (%)
ResNet50	0 (baseline)	ConvBlock1-5	×	75.61
	1	ConvBlock1-4	ConvBlock5	76.98
	2	ConvBlock1-3	ConvBlock4-5	77.79
	3	ConvBlock1-2	ConvBlock3-5	79.48
	4	ConvBlock1	ConvBlock2-5	77.42
	5	×	ConvBlock1-5	78.23
VGG16	6 (baseline)	ConvBlock1-5	×	66.01
	7	ConvBlock1-4	ConvBlock5	68.14
	8	ConvBlock1-3	ConvBlock4-5	73.26
	9	ConvBlock1-2	ConvBlock3-5	72.97
	10	ConvBlock1	ConvBlock2-5	70.53
	11	×	ConvBlock1-5	72.69

Note: "×" ……



(**a**) resized_frame0 (**b**) ConvBlock1

(**c**) ConvBlock2 (**d**) ConvBlock3

(**e**) ConvBlock4 (**f**) ConvBlock5

Figure 13. Feature visualization results.

Figure 14 shows the visualization effect for 64 convolution kernels of ConvBlock1 in the ResNet50 network. It can be observed that different convolution kernels can obtain different image features, which ensures that the FT-ResNet50 model has strong feature acquisition ability.

Figure 14. Filter visualization results.

4. Discussion

The experimental results in Table 5 show that the proposed sample expansion strategy had a positive impact on improving the accuracy of forest fire identification. The expansion of the sample size made the training data more diversified, which can reduce the field transfer of training and testing to a certain extent. Therefore, compared with the dataset before the sample expansion, the expanded dataset achieved higher forest fire recognition accuracy.

The choice of loss function also affects the accuracy of forest fire identification. The experimental results in Tables 6 and 7 show that, compared with the traditional cross-entropy loss function, the Focus Loss function focused more on the training of difficult samples, which helped to improve the learning ability of the network.

The experimental results shown in Table 8 indicate that different depths of the ResNet network affect the recognition accuracy and operational performance of the model. With the deepening of network layers, although the accuracy of forest fire identification is improved, the time consumption of operations also increases. After weighing the model complexity, training cost, test accuracy and other factors, this study finally selected ResNet 50 as the backbone network for feature extraction.

The selection of activation function and optimizer affects forest fire recognition accuracy. The experimental results shown in Table 9 show that better recognition results can be obtained by fine-tuning the convolution block with Mish as the activation function and Adam as the optimizer. This is because the Mish activation function can effectively improve the gradient loss in network training. The Adam optimizer can avoid the model falling into local optimization during training.

In this study, six transfer learning schemes were designed for each network model, given in the second column of Table 10. Specifically, “Scheme 0” represents the base scheme baseline in which the pre-training weights of all five convolutional blocks were fixed, and the network did not require fine-tuning during the training; “Scheme 1” to “Scheme 4” indicate the fine-tuning scheme from deep to shallow layers respectively, which gradually unlocked the fine-tuning operation of the deep layer network; “Scheme 5” indicates the fine-tuning of all the pre-training parameters. Observing the results in Table 10, we can see that the ReNet50 network achieved the highest recognition accuracy

by fixing the first two and fine-tuning the last three ConvBlocks, while the VGG16 network achieved a higher detection result by fixing the first three and fine-tuning the last two ConvBlocks. The accuracy of "Scheme 1" and of "Scheme 7", which only fine-tuned the last convolutional block, is lower than that of schemes that fine-tuned the last two convolutional blocks; e.g., in the ResNet architecture "Scheme 1" was reduced by 0.81 percentage points compared with "Scheme 2", and by 2.5 percentage points compared with "Scheme 3". This is because features acquired by deeper networks are often abstract and have strong category correlation, therefore, fixed parameter information inhibits to a certain extent the ability of the network to capture discriminant information for the current task. In other words, the images in the ImageNet dataset tend to depict natural landscapes, animals, plants, etc., with low applicability after training on these images for transferring high-level semantic information obtained by the backbone network to the current forest fire image task. Low-level information such as edge, texture, color, etc., is often universal, and no matter the type of image it will cover similar features to a certain extent. Transferring and fixing trained shallow network weights to the proposed model will help improve the network convergence performance and prevent the deterioration of training. However, the parameters before which convolution blocks are to be fixed need specific analysis for different problems, so it is difficult to identify from the theoretical level the specific level most beneficial to the migration effect, and the situation is different for different models. For example, for the current forest fire image recognition problem, it is better to use ResNet50 to fix the first two convolution blocks, and VGG16 to fix the first three convolution blocks. "Scheme 5" and "Scheme 11" provide a training scheme for fine-tuning all migration parameters. It can be seen from the results that this training method achieved relatively good performance. "Scheme 5" and "Scheme 11" are poor than "Scheme 3" and "Scheme 8" because fine tune all parameters are prone to large fluctuations, especially in the initial stage, the training of the layered transmission characteristics influence each other, more "error" may occur between to optimize parameters of cumulative phenomenon, resulting in instability in the process of training. Finally, we selected "Scheme 3" as the fine-tuning method for the transfer learning strategy in the subsequent experimental process.

In order to verify that the FT-ResNet50 model can effectively extract image features, this paper visually displays the features captured by FT-ResNet50. The results in Figure 13 show that with the increase of network depth, the extracted feature level also increases. Low-level features initially extracted from Convblock1 and Convblock2, such as edge, texture and color, are transformed in Convblock3 and subsequent convolution blocks into high-level abstract features with stronger task relevance. The results shown in Figure 14 show that different convolution kernels also help to obtain different features of the image. The more types of convolution kernels, the better the feature extraction ability. Therefore, the FT-ResNet50 model proposed in this paper can extract more abundant features from fire images, and can better improve the accuracy of forest fire classification by combining low-level edge information with high-level semantic information.

5. Conclusions

Forest fire recognition based on image processing is an important method to assist with the early detection of forest fire. Deep learning is an important research direction for forest fire identification. Taking the improved ResNet50 as the backbone framework, this paper proposes a forest fire identification model, FT-ResNet50. FT-ResNet50 combines the mixup-based augmentation method with the traditional sample augmentation method to increase the number of training samples, thereby improving the generalization ability of the model. Considering the effects on the accuracy of forest fire identification of loss function, backbone network, network depth and transfer learning strategy, this study determined the optimal configuration of the model. At the same time, this paper also discusses the influence on forest fire recognition of different

block parameter convolution adjustments. The experimental results show that, based on UAV images with limited labeled samples, the FT-ResNet50 model proposed in this paper can realize high-performance forest fire recognition tasks.

References

1. Mahmoud, M.A.I.; Ren, H. Forest fire detection and identification using image processing and SVM. *J. Inf. Processing Syst.* **2019**, *15*, 159–168.
2. Alkhatib, A.A.A. A review on forest fire detection techniques. *Int. J. Distrib. Sens. Netw.* **2014**, *10*, 597368.
3. Xu, R.; Lin, H.; Lu, K.; Cao, L.; Liu, Y. A forest fire detection system based on ensemble learning. Forests **2021**, *12*, 217.
4. Fernandes, A.M.; Utkin, A.B.; Lavrov, A.V.; Vilar, R.M. Development of neural network committee machines for automatic forest fifire detection using lidar. *Pattern Recognit.* **2004**, *37*, 2039–2047.
5. Barmpoutis, P.; Papaioannou, P.; Dimitropoulos, K.; Grammalidis, N. A review on early forest fire detection systems using optical remote sensing. *Sensors* **2020**, *20*, 6442.
6. Zhang, F.; Zhao, P.; Xu, S.; Wu, Y.; Yang, X.; Zhang, Y. Integrating multiple factors to optimize watchtower deployment for wildfifire detection. *Sci. Total Environ*. **2020**, *737*, 139561.
7. Zhang, F.; Zhao, P.; Thiyagalingam, J.; Kirubarajan, T. Terrain-inflfluenced incremental watchtower expansion for wildfifire detection. *Sci. Total Environ*. **2018**, *654*, 164–176.
8. Çetin, A.E.; Dimitropoulos, K.; Gouverneur, B.; Grammalidis, N.; Günay, O.; Habiboğlu, Y.H.; Töreyin, B.U.; Verstockt, S. Video fire detection–review. *Digit. Signal Processing* **2013**, *23*, 1827–1843.
9. Mahmoud, M.A.I.; Ren, H. Forest fire detection using a rule-based image processing algorithm and temporal variation. *Math. Probl. Eng.* **2018**, 2018(7):1-8.
10. Sudhakar, S.; Vijayakumar, V.; Kumar, C.S.; Priya, V.; Ravi, L.; Subramaniyaswamy, V. Unmanned Aerial Vehicle (UAV) based Forest Fire Detection and monitoring for reducing false alarms in forest-fires. *Comput. Commun.* **2020**, *149*, 1–16.
11. Wu, H.; Li, H.; Shamsoshoara, A.; Razi, A.; Afghah, F. Transfer learning for wildfire identification in UAV imagery. In Proceedings of the 2020 54th Annual Conference on Information Sciences and Systems (CISS), Princeton, NJ, USA, 18–20 March 2020; pp. 1–6.
12. Yuan, C.; Zhang, Y.; Liu, Z. A survey on technologies for automatic forest fire monitoring, detection, and fighting using unmanned aerial vehicles and remote sensing techniques. *Can. J. For. Res.* **2015**, *45*, 783–792.
13. Celik, T.; Ozkaramanli, H.; Demirel, H. Fire pixel classification using fuzzy logic and statistical color model. In Proceedings of the 2007 IEEE International Conference on Acoustics, Speech and Signal Processing-ICASSP'07, Honolulu, HI, USA, 15–20 April 2007; pp. 1: I-1205–I-1208.
14. Jiao, Z.; Zhang, Y.; Xin, J.; Mu, L.; Yi, Y.; Liu, H.; Liu, D. A deep learning based forest fire detection approach using UAV and YOLOv3. In Proceedings of the 2019 1st International Conference on Industrial Artificial Intelligence (IAI), Shenyang, China, 23-27 July 2019; pp. 1–5.
15. Anh, N.D.; Van Thanh, P.; Lap, D.T.; Khai, N.T.; Van An, T.; Tan, T.D.; Dinh, D.N. Efficient Forest Fire Detection using Rule-Based Multi-color Space and Correlation Coefficient for Application in Unmanned Aerial Vehicles. *KSII Trans. Internet Inf. Syst. (TIIS)* **2022**, *16*, 381–404.
16. Yuan, C.; Liu, Z.; Zhang, Y. UAV-based forest fire detection and tracking using image processing techniques. In Proceedings of the 2015 International Conference on Unmanned Aircraft Systems (ICUAS), Denver, CO, USA, 9–12 June 2015; pp. 639–643.
17. AlZu'bi, S.; Jararweh, Y. Data fusion in autonomous vehicles research, literature tracing from imaginary idea to smart surrounding community. In Proceedings of the 2020 Fifth International Conference on Fog and Mobile Edge Computing (FMEC), Paris, France, 20–23 April 2020; pp. 306–311.
18. Elbes, M.; Almaita, E.; Alrawashdeh, T.; Kanan, T.; AlZu'bi, S.; Hawashin, B. An indoor localization approach based on deep learning for indoor location-based services. In Proceedings of the 2019 IEEE Jordan International Joint Conference on Electrical Engineering and Information Technology (JEEIT), Amman, Jordan, 9–11 April 2019; pp. 437–441.
19. Aqel, D.; Al-Zubi, S.; Mughaid, A.; Jararweh, Y. Extreme learning machine for plant diseases classification: A sustainable approach for smart agriculture. *Clust. Comput.* **2022**, *25*, 2007–2020.
20. Hu, Y.; Zhan, J.; Zhou, G.; Chen, A.; Cai, W.; Guo, K.; Hu, Y.; Li, L. Fast forest fire smoke detection using MVMNet. *Knowl. -Based Syst.* **2022**, *241*, 108219.
21. Guan, Z.; Min, F.; He, W.; Fang, W.; Lu, T. Forest fire detection via feature entropy guided neural network. *Entropy* **2022**, *24*, 128.
22. Li, T.; Zhang, C.; Zhu, H.; Zhang, J. Adversarial Fusion Network for Forest Fire Smoke Detection. *Forests*, **2022**, *13*, 366.
23. Fan, R.; Pei, M. Lightweight Forest Fire Detection Based on Deep Learning. In Proceedings of the 2021 IEEE 31st International Workshop on Machine Learning for Signal Processing (MLSP), Gold Coast, Australia, 25–28 October 2021; pp. 1–6.
24. Guede-Fernández, F.; Martins, L.; Almeida, R.V.; Gamboa, H.; Vieira, P. A deep learning based object identification system for forest fire detection. *Fire* **2021**, *4*, 75.
25. Weiss, K.; Khoshgoftaar, T.M.; Wang, D.D. A survey of transfer learning. *J. Big Data* **2016**, *3*, 1–40.
26. Zamir, A.; Sax, A.; Shen, W.; Guibas,C.; Savarese,S. Taskonomy: Disentangling Task Transfer Learning. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 2018; pp. 3712–3722. .

27. Shamsoshoara, A.; Afghah, F.; Razi, A.; Zheng, L.; Fulé, P.Z.; Blasch, E. Aerial Imagery Pile burn detection using Deep Learning: The FLAME dataset. *Comput. Netw.* **2021**, *193*, 108001.
28. Zhang, H.; Cisse, M.; Dauphin, Y.N.; Lopez-Paz, D. mixup: Beyond empirical risk minimization. arXiv **2017**, preprint arXiv:1710.09412.
29. Simonyan, K.; Zisserman, A. Very deep convolutional networks for large-scale image recognition. arXiv **2014**, preprint arXiv:1409.1556.
30. 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.
31. Deng, J.; Dong, W.; Richard, S.; Li, L.J.; Li, K.; Li, F.-F. ImageNet：A Largescale Hierarchical Image Database. In Proceedings of the IEEE Computer Vision and Pattern Recognition (CVPR), Miami, FL, USA, 20–25 June 2009; pp. 248–255.
32. Li, C.; Zhang, D.; Du, S.; Zhu, Z.; Jia, S.; Qu, Y. A Butterfly Detection Algorithm Based on Transfer Learning and Deformable Convolution Deep Learning. *Acta Autom. Sin.* **2019**, *45*, 1772–1782.
33. Zhang, Z. Improved adam optimizer for deep neural networks. In Proceedings of the 2018 IEEE/ACM 26th International Symposium on Quality of Service, Banff, AB, Canada, 4–6 June 2018; pp. 1–2.
34. Duchi, J.; Hazan, E.; Singer, Y. Adaptive subgradient methods for online learning and stochastic optimization. *J. Mach. Learn. Res.* **2011**, 257-269.
35. Tieleman, T.; Hinton, G. Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude. *COURSERA: Neural Netw. Mach. Learn.* **2012**, *4*, 26–31.
36. Kingma, D.P.; Ba, J. Adam: A method for stochastic optimization. arXiv **2014**, preprint arXiv:1412.6980.
37. Yan, Q.; Yang, B.; Wang, W.; Wang, B.; Chen, P.; Zhang, J. Apple Leaf Diseases Recognition Based on An Improved Convolutional Neural Network. *Sensors* **2020**, *20*, 3535.
38. Isola, P.; Zhu, J.Y.; Zhou, T.; Efros, A.A. Image-to-Image Translation with Conditional Adversarial Networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 1125–1134.
39. Lin, T.Y.; Goyal, P.; Girshick, R.; He, K.; Dollár, P. Focal Loss for Dense Object Detection. IEEE Transactions on Pattern Analysis & Machine Intelligence, 2017, PP(99):2999-3007.
40. Misra, D. Mish: A Self Regularized Non-Monotonic Neural Activation Function. arXiv 2019, preprint arXiv:1908.08681.