symmetry

MDPI

Article

A Recognition Method of Ancient Architectures Based on the Improved Inception V3 Model

Xinyang Wang [1,2,*], Jiaxun Li [1], Jin Tao [3,*], Ling Wu [1], Chao Mou [1,2], Weihua Bai [4], Xiaotian Zheng [1], Zirui Zhu [1] and Zhuohong Deng [1,5]

[1] School of Information Science and Technology, Beijing Forestry University, Beijing 100083, China
[2] Engineering Research Center for Forestry-Oriented Intelligent Information Processing of National Forestry and Grassland Administration, Beijing 100083, China
[3] School of Architecture, South China University of Technology, Guangzhou 510641, China
[4] School of Computer Science, Zhaoqing University, Zhaoqing 526061, China
[5] School of Electronic Information and Electrical Engineering, Shanghai Jiao Tong University, Shanghai 200240, China
* Correspondence: wxyyuppie@bjfu.edu.cn (X.W.); arjtao@scut.edu.cn (J.T.)

Abstract: Traditional ancient architecture is a symbolic product of cultural development and inheritance, with high social and cultural value. An automatic recognition model of ancient building types is one possible application of asymmetric systems, and it will be of great significance to be able to identify ancient building types via machine vision. In the context of Chinese traditional ancient buildings, this paper proposes a recognition method of ancient buildings, based on the improved asymmetric Inception V3 model. Firstly, the improved Inception V3 model adds a dropout layer between the global average pooling layer and the SoftMax classification layer to solve the overfitting problem caused by the small sample size of the ancient building data set. Secondly, migration learning and the ImageNet dataset are integrated into model training, which improves the speed of network training while solving the problems of the small scale of the ancient building dataset and insufficient model training. Thirdly, through ablation experiments, the effects of different data preprocessing methods and different dropout rates on the accuracy of model recognition were compared, to obtain the optimized model parameters. To verify the effectiveness of the model, this paper takes the ancient building dataset that was independently constructed by the South China University of Technology team as the experimental data and compares the recognition effect of the improved Inception V3 model proposed in this paper with several classical models. The experimental results show that when the data preprocessing method is based on filling and the dropout rate is 0.3, the recognition accuracy of the model is the highest; the accuracy rate of identifying ancient buildings using our proposed improved Inception V3 model can reach up to 98.64%. Compared with other classical models, the model accuracy rate has increased by 17.32%, and the average training time has accelerated by 2.29 times, reflecting the advantages of the model proposed in this paper. Finally, the improved Inception V3 model was loaded into the ancient building identification system to prove the practical application value of this research.

Keywords: deep learning; Inception V3; transfer learning; ancient architecture classification; dropout layer

check for updates

Citation: Wang, X.; Li, J.; Tao, J.; Wu, L.; Mou, C.; Bai, W.; Zheng, X.; Zhu, Z.; Deng, Z. A Recognition Method of Ancient Architectures Based on the Improved Inception V3 Model. *Symmetry* **2022**, *14*, 2679. https://doi.org/10.3390/sym14122679

Academic Editors: Dumitru Baleanu and Carlo Cattani

Received: 16 October 2022
Accepted: 12 December 2022
Published: 18 December 2022

1. Introduction

Chinese traditional ancient architecture bears witness to China's excellent historical culture and glorious historical achievements; it carries the historical development of the Chinese nation and the local area and has irreplaceable cultural, social, and artistic value. Most of this historical architecture is widely distributed and is scattered in remote villages; it gradually disappeared with urban development and change, bringing irreparable losses to China's history, culture, economy, etc. Therefore, it is very necessary to study and protect Chinese traditional ancient architecture. Different ancient architectural communities represent different cultures and also need different protection methods. Among them, the accurate recognition of ancient architecture is an important prerequisite for formulating targeted ancient architecture protection measures. Traditional methods, such as relying on human experience and judgment, are inefficient, error-prone, and highly dependent on personal experience and professional skills; thus, they are not suitable for large-scale promotion and application. Therefore, the development and maturity of deep learning, image processing, and other technologies provide new ideas and efficient means for the recognition and classification of ancient architecture. The combination of deep learning and other technologies with the identification and protection of ancient architecture can give better play to the advantages of cutting-edge technologies and can push the protection of ancient architecture to a new and higher level [1].

The feature extraction of traditional ancient architecture mainly depends on the architecture's color and structure. For some ancient architectures with a roughly similar appearance, due to the influence of different factors, such as light angle, shooting angle, and the building's weathering degree during the image acquisition process, the recognition efficiency and accuracy will be greatly reduced by labor that has weak professional knowledge. In view of this issue, one common method is to extract features from ancient architectural images, calculate the distance between the extracted features and the features in the scale map, and then use this as the basis for classification to judge the types of buildings. For example, Wang et al. [2] proposed a building recognition algorithm that combines local features with shape contour-matching, which has a good recognition effect in terms of different angles, different scales, and different lighting conditions. Wu [3] proposed an automatic classification algorithm for ancient Buddhist buildings, based on local features, which can reasonably segment Buddhist building images and can finally achieve a good recognition result. Hasan M et al. [4] used four different feature detection methods to realize the identification of the architectural age of three types of era for the ancient and heritage buildings of the Indian subcontinent. Zhang et al. [5] proposed an ancient architecture image-annotation method based on the visual attention mechanism and the graph convolution network, which can effectively improve the annotation accuracy of ancient Chinese architecture and enrich the semantic information. Yang Song et al. [6] proposed a building recognition method based on the improved histogram features of gradient direction. Freeman et al. [7] proposed a direction-controlled filter, which shows great advantages in terms of image edge detection and texture analysis.

The deep convolutional neural network is another effective means of building-type recognition. In terms of the convolution layer structure, such a network has a variety of structures, such as conventional convolution, atrous convolution [8], depthwise separable convolution [9], etc. Classical convolutional neural network structures mainly include LeNet [10], AlexNet [11], VGGNet [12], ResNet [13], InceptionNet [14], and so on. LeNet [10] is the earliest convolutional network that was used for image classification tasks, including 7 convolutional layers; AlexNet [11] expands the network structure to 8 layers, becoming much larger in scale than LeNet, and introduces the ReLU activation function and image enhancement; the symmetric convolutional network, VGGNet [12], is increased to 16 layers and 19 layers, and is deeper than AlexNet but is simpler in form; the symmetric convolutional network, ResNet [13], is 8 times deeper than VGGNet but has lower complexity and strong characterization ability, which solves the problem of gradient disappearance through a residual structure. Compared with the way that VGGNet vertically stacks the convolution layer, InceptionNet [14] adopts a completely different approach, i.e., it improves the network capacity by horizontally increasing the network width and uses "global average pooling + full connection layer" to greatly reduce the parameter scale. In recent years, with the continuous development and application of convolutional neural networks, various classical network models have been constantly improved and can be applied to many different fields. For example, in the AlexNet network, some scholars changed the network framework by increasing the network depth, replacing the FC layer, and changing the intranet network by adding a BN layer and LRN layer, which ultimately improved the network's accuracy and training speed [15]. Abhronil Sengupta et al. [16] proposed a new weighting normalization technique to ensure that the actual spiking neural network operation was in the loop during the conversion phase. This work, which was aimed at optimizing the ratio of synaptic weights to neuronal firing thresholds, ultimately demonstrated the effectiveness of the architecture for complex visual recognition problems; it can be used in the VGG and ResNet architectures, bringing better accuracy than the most advanced techniques. Ayesha Younis et al. [17] used the VGG 16 architecture as the main network to generate a convolutional feature map. An effective method using MRI to detect brain tumors was developed and improved, which was superior to the current traditional brain tumor detection method. It can be seen that all kinds of convolutional neural networks are constantly improving, progressing, and developing.

However, different convolution layer structures define the different feature extraction methods, so that the convolution neural network can fully extract the image features. In terms of neural network structure, the connection mode between layers and the size of the convolution kernel are important factors affecting the accuracy of model recognition. Previous studies have shown that using a larger convolution kernel will achieve better results [18]. It can be seen that the convolutional neural network has more room for development in building-type recognition. For example, Professor Xiao [19], from the South China University of Technology, proposed an automatic recognition and classification algorithm based on ResNet152, which is applicable to a large number of traditional village status maps; the overall recognition accuracy reached 83.4%. Guo et al. [20] applied a convolutional neural network to building style image classification, designed a shallow classification model and deep classification model, respectively, from aspects of network structure design and parameter optimization, and improved model generalization capability via DropConnect. Building identification methods based on deep convolutional neural networks have high identification efficiency and accuracy, which is helpful for improving the archiving and management efficiency of ancient architecture [21]. In addition, in recent years, transfer learning has been widely used in the image classification tasks of convolutional neural networks. By retaining the weight and bias of the original model, some layers of the neural network are trained to improve model training efficiency [22]. For example, Wang et al. [23] used the pre-trained VGG16 network for transfer learning, to improve the recognition effect. At the same time, they proposed an adaptive feature fusion method to fuse the features extracted from the VGG16 network for prediction and obtained a satisfactory recognition effect. However, the above methods have some shortcomings, such as too-deep network layers, the huge scale of model training parameters, a slow training process, etc. Moreover, due to the small number and scattered distribution of ancient architectures, it is easy to experience an over-fitting phenomenon caused by insufficient training samples, and there is a lack of recognition models specifically intended for ancient architectures.

In view of the above reasons, this paper adopts the improved Inception V3 [14] model to identify ancient architectures. The Inception V3 model decomposes large convolution kernels into small convolution kernels and has excellent parallel convolution ability and strong model expression ability. It supports high-performance computing, based on dense matrices, and can process more and richer spatial features. However, if the Inception V3 model is directly applied to ancient architecture type identification, it will lead to overfitting, low training efficiency, and other problems, so the model needs to be further improved. This paper fully combines the advantages of the Inception V3 model, such as decomposing large convolution kernels into small convolution kernels with strong expression ability, supporting parallel convolution, and the advantages of the dropout layer's strong ability to prevent overfitting, to propose an improved Inception V3 model with a dropout layer. In this paper, comparative experiments are carried out, after which the best and most effective method is used to preprocess the self-built ancient architecture dataset, then transfer learning technology is introduced. Finally, the overfitting phenomenon in the model training process is alleviated, and the model training efficiency is improved; in addition, the model calculation efficiency is increased, and the model recognition accuracy is improved. It provides a useful reference scheme for the accurate identification of traditional Chinese ancient architecture, based on deep learning technology.

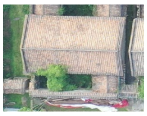
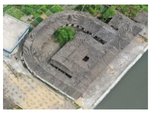
The rest of this paper is organized as follows. Section 2 describes and pre-processes the ancient building dataset. Section 3 introduces the improved Inception V3 model. Section 4 explains our methodology. Section 5 verifies the effectiveness of our proposed method by means of experiments. Section 6 concludes this paper.

2. Description and Pre-processing of the Ancient Building Dataset

2.1. Dataset of Ancient Architecture

At present, there is no professional ancient architecture dataset for China. This paper adopts the dataset for traditional residential ancient architecture in the Guangdong Province of China as the research and identification object, which was constructed by the team of the South China University of Technology over many years. This dataset includes six types of ancient architecture, including the square-shaped Tulou, Canton ancestral hall, Leiqiong folk house, San-Jian Liang-lang folk house, Hakka walled village, and round Tulou. The images of each architecture type are shown from different viewpoints, comprising the top view, overhead view, internal view, and external view. Examples of the datasets are shown in Table 1. The symbol "–" indicates empty data.

Table 1. Examples of the dataset.

	Top View	**Overhead View**	**Internal View**	**External View**
Square-shaped Tulou			–	
Canton ancestral hall				
Leiqiong folk house	–		–	
San-Jian Liang-lang folk house			–	
Hakka walled village				
Round Tulou				

The number of samples from the different views of each type of ancient architecture in the dataset is shown in Table 2.

In the process of model training, the total samples are randomly allocated; 80% of these are training sets, and the remaining 20% are test sets. Finally, 4418 training images and 1095 test images are obtained. Due to the interference of various factors in the process of data collection, certain views of some buildings are not collected. When dividing the training set and test set, the data need to be disordered; otherwise, this will result in an insufficient training process and weak model generalization ability. Saving the training set, the training set labels, test set, and test set labels separately ensures independence between the training set and test set and improves the model's training efficiency.

Table 2. The number of samples from the different views for each type of architecture.

	Top View	Overhead View	Internal View	External View
Square-shaped Tulou	55	404	0	227
Canton ancestral hall	286	899	82	528
Leiqiong folk house	0	65	0	113
San-Jian Liang-lang folk house	225	561	0	280
Hakka walled village	307	422	53	198
Round Tulou	250	322	109	127

2.2. Pre-Processing of the Datasets

Since the input of the Inception V3 model is a square image, sized 299 × 299 pixels, most of the images in the self-built dataset adopted in this paper have different aspect ratios. For this reason, it is necessary to pre-process datasets to meet the input requirements of the model. Optional image data pre-processing methods include: (1) directly modifying the image aspect ratio to make it a square image, and then changing the image size to the required input size; (2) cropping the image center according to the short side; (3) filling the image according to the long side. The results of image pre-processing by directly modifying the image size, cropping, and filling are shown in Figures 1–3, respectively. In Section 5, the effects of using three different data pre-processing methods will be compared and analyzed via experiments.

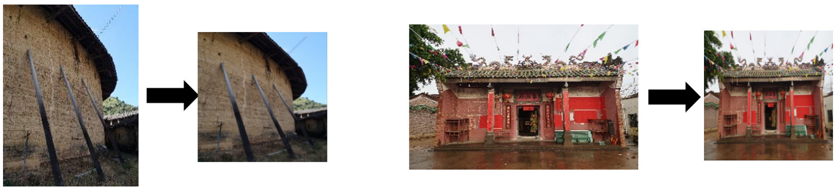


Figure 1. The pre-processing method of directly modifying the image size.

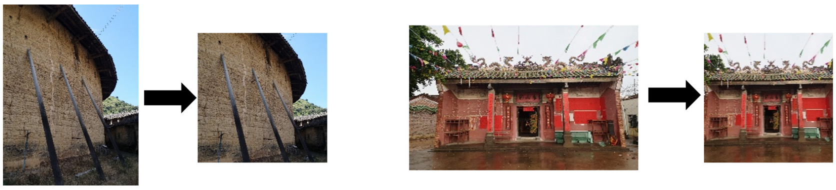


Figure 2. The pre-processing method of cropping the center part of the image.

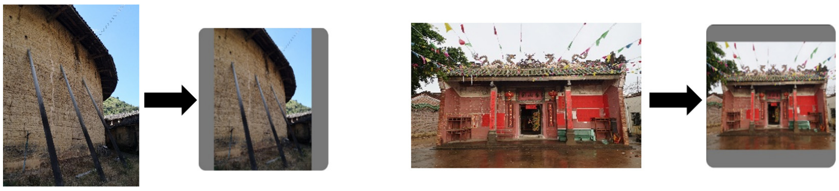


Figure 3. The pre-processing method of filling out the image.

3. The Improved Model, Based on Inception V3

3.1. Analysis of the Inception V3 Model

The most direct way to improve the network performance is to increase the depth and width of the network model. However, at the same time, it will bring two problems: (1) the increase in the network depth and width will produce more parameters, which will lead to overfitting; (2) this will significantly increase the amount of calculation. The above problems can be solved by introducing sparse features and converting the full connection layer to sparse connections, but the existing models cannot effectively support the fast operation of non-uniform sparse data. In order to support both the sparse feature at the filter level and the sort of high-performance computing that makes full use of dense matrices, the Inception network family of software, based on the multi-branch module or convolution decomposition parallel structure, is an ideal choice that decomposes large-scale convolution into multiple small-scale convolutions through Inception modules, significantly reduces the computational complexity of parameters, and achieves the balance of network width and depth while retaining the same feature expression capability. The sparse matrix is decomposed into a dense matrix calculation to accelerate the network convergence speed.

The inception network family is based on the idea of a network within a network, and is asymmetric in structure. The Inception V1 model [24] proposes a modular multi-branch network structure that reduces the computation of the convolution kernel and achieves more accurate classification results by extracting features on different scales. The Inception V2 model [25] proposes batch normalization on the basis of V1, which improves the speed of early training and effectively alleviates the problems of gradient disappearance and overfitting. The Inception V3 model [14] further proposes decomposable convolution on the basis of V1 and V2 and decomposes the large convolution into a small convolution and asymmetric convolution. Compared with the Inception V2 model, the parameter number of the Inception V3 model is reduced by 1.38 times when using smaller convolution kernels to further deepen the network. Under the premise of the same feature map size and receptive field, its network depth is deepened, while the training speed is improved. Inception V3 plays a connecting role within the whole Inception network family. The convolution decomposition method proposed by V3 provides a new idea for the subsequent CNN model. The convolution and merging process of the Inception modules is shown in Figure 4.

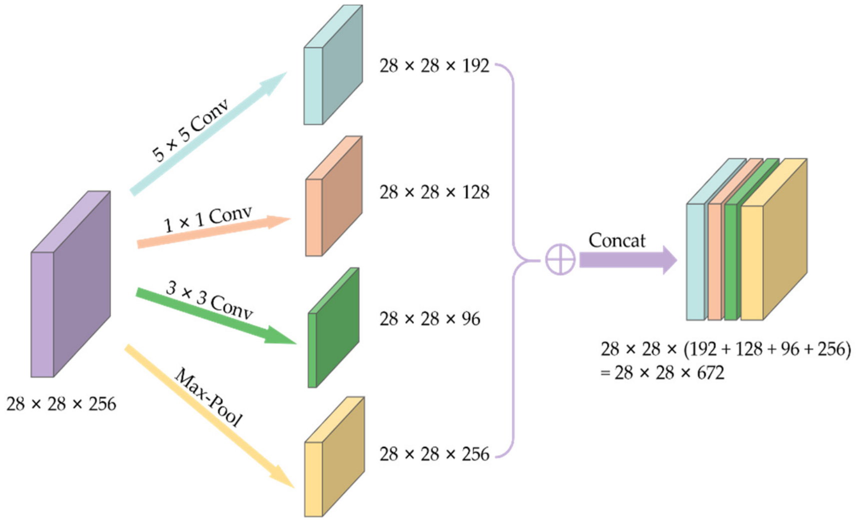


Figure 4. The convolution and merging process of the Inception modules.

3.2. Improved Inception V3 Model

The number of preserved ancient architecture examples is small, and the distribution is generally scattered. It is costly in terms of labor and the economy to capture sufficient high-quality image datasets of ancient buildings, which is a problem for the automatic recognition of ancient buildings using deep learning technologies, owing to the overfitting problem caused by insufficient training samples. Dropout [26,27], proposed by Hinton in 2012, is a method that is applied regularly in the field of computer vision and is effective in preventing model overfitting. In each training process, dropout temporarily "drops" the neural network unit from the network, according to a certain probability, and alleviates overfitting by reducing the scale of the network parameters required for each training iteration. The specific process is as follows:

(1) Randomly stop some neurons without changing the input and output;
(2) Put the input into the network for forward propagation, and the loss result still uses backpropagation in the network. After completing the forward propagation of training samples and the backpropagation of loss, the parameters of the working neurons are optimized, while the parameters of the stopped neurons remain unchanged;
(3) Repeat the process of (1) and (2) until the loss function stabilizes.

The calculation formula of the dropout algorithm is as follows:

$$r_j^{(l)} \sim Bernoulli(p) \tag{1}$$

$$y_i^{(l+1)} = f(w_i^{(l+1)} \cdot I(x) \cdot r_j^{(l)} + b_i^{(l+1)}) \tag{2}$$

where $r_j^{(l)}$ is a probability vector containing only 0 and 1 elements and obeying a Bernoulli distribution. $w_i^{(l+1)}$ and $b_i^{(l+1)}$ represent the weight and bias of the *ith* node in the next layer of layer l, respectively, $I(x)$ represents the feature vector extracted through the Inception V3 network, and $y_i^{(l+1)}$ represents the output after passing the activation function, $f()$. The working principle of dropout in the model is shown in Figure 5.

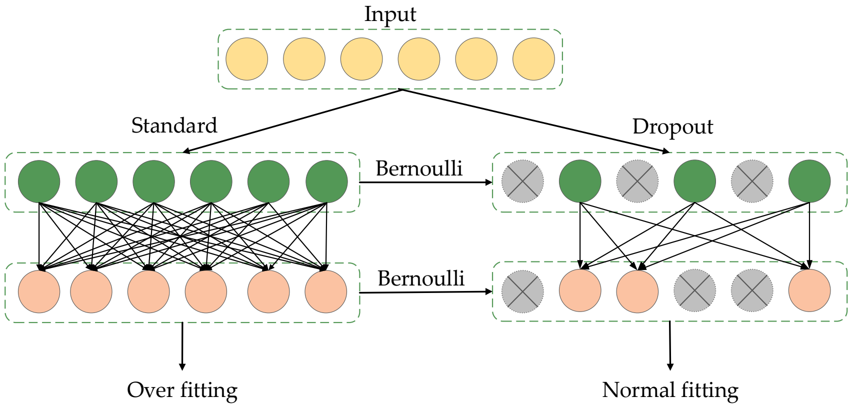


Figure 5. The working principle of dropout.

Inception V3 has innovated four new structures of Inception, abandoned the original bottom auxiliary classifier, and retained only one auxiliary classifier at the middle level. It has formed a new Inception V3 model with wider, deeper, better performance and stronger expression ability, and has also enhanced computing efficiency. For Inception Module A, the original 5 × 5 convolution kernel is replaced with two 3 × 3 convolution kernels, with the second 3 × 3 convolution kernel being a full connection layer and equivalent to one 5 × 5 convolution kernel. It retains the receptive field at 5 × 5 but greatly reduces the parameter amount. For a convoluted layer with C filters, the parameters required by different convolution kernels are shown in Table 3.

Table 3. Parameters required by different convolution kernels.

Convolution Kernel and Quantity	Required Parameter Quantity
One 5 × 5 Conv	$(H \times W \times C) \times (5 \times 5 \times C) = 25\,HWC^2$
Two 3 × 3 Convs	$\mathbf{2 \times (H \times W \times C) \times (3 \times 3 \times C) = 18\,HWC^2}$

Here, *H* represents the height of the input images, *W* represents the width of the input images, and *C* represents the number of filters. As can be seen from Table 3, the parameter amount using two 3 × 3 convolution kernels can be reduced by 28%. For Inception Module

B, using a pair of 1 × N and N × 1 asymmetric convolution kernels on the feature map of the middle size in order to replace one N × N convolution kernel can also reduce the number of parameters and improve the calculation efficiency. For Inception Module C, the method of expanding along the width instead of along the depth is adopted, which can increase the representation dimension and expand the feature dimension. For the down-sampling module, the V3 model adopts a parallel method, which can not only reduce the grid size but also ensure the increase in channels, so as to achieve the goal of both reducing the amount of computation and expanding the dimensions.

Therefore, in order to solve the problem of overfitting, which may be caused by the insufficient sample size of the ancient architecture dataset, and to improve the recognition effect, this paper proposes an improved Inception V3 model, combining the advantages of the Inception V3 model (such as strong expression ability and rich spatial features, supportive of parallel convolution) and the strong ability of dropout in terms of preventing overfitting. The improved Inception V3 model structure is shown in Figure 6.

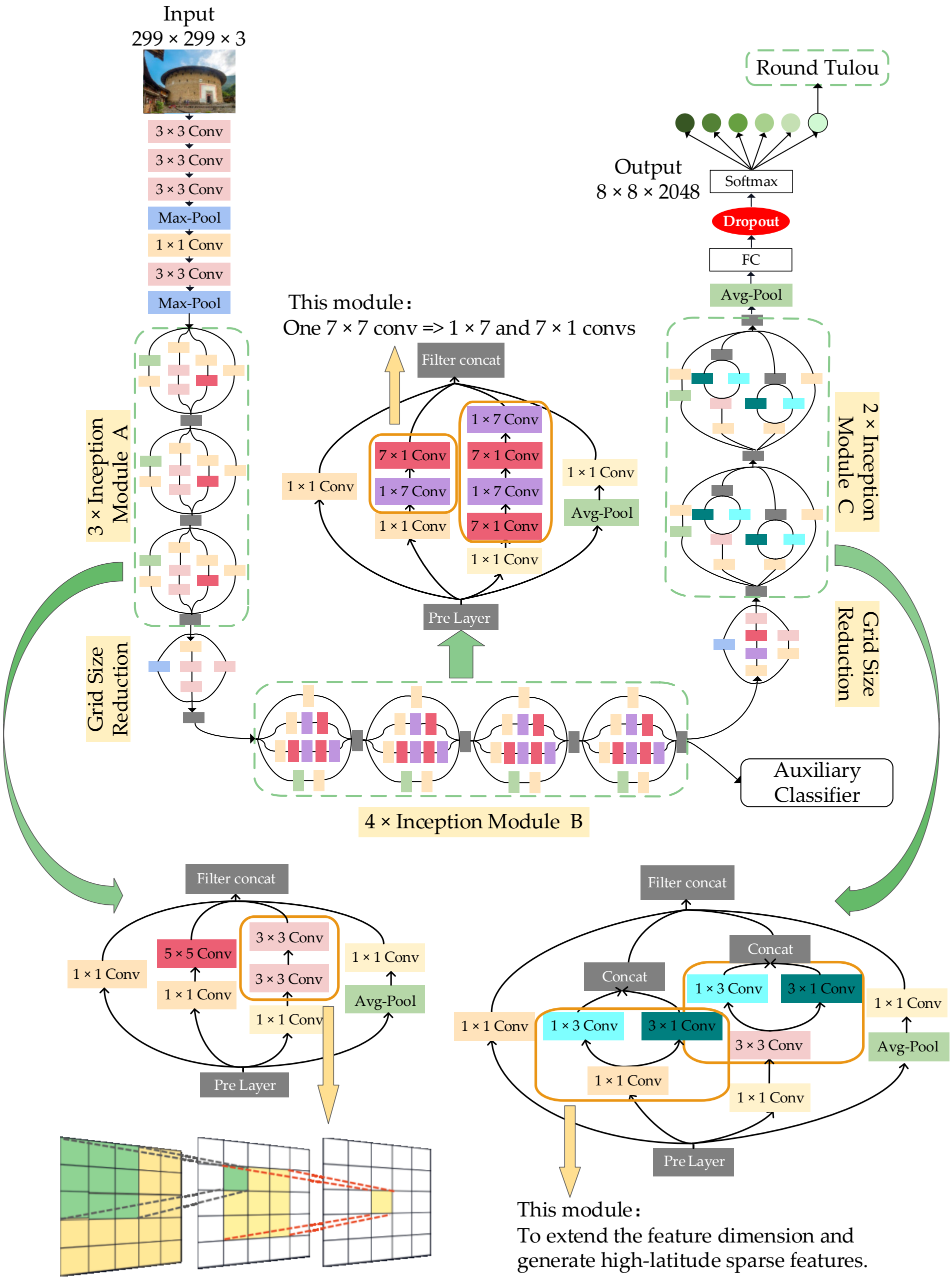


Figure 6. Improved Inception V3 model structure.

The above network model structure diagram shows the complete process from input to output of the recognition results of ancient architectural images in this model. We assume that the size of the ancient architecture training set in this paper is m, the dimension of the feature vector of the input sample X is (n_x, m), the input sample is $X \in R^{n_x \times m}$, and the output is $Y \in R^{1 \times m}$. Taking layer l as an example, the vectorization process of the forward propagation of the model is as follows:

$$Z^{[l]} = w^{[l]} \cdot A^{[l-1]} + b^{[l]} \tag{3}$$

$$A^{[l]} = R^{[l]}(Z^{[l]}). \tag{4}$$

where $A^{[0]} = X$, $A^{[l-1]}$ represents the input of layer l as the output of the previous layer, $A^{[l]}$ represents the output of layer l, $w[l]$ and $b[l]$ represent the weight and bias of layer l respectively, and $R^{[l]}()$ is the output value after activation using the *ReLU* nonlinear activation function. The cross-entropy cost is calculated after obtaining the output value, and is formulated as follows:

$$\text{cost} = -\frac{1}{m} \sum_{i=1}^{m} ((y^{(i)} \log(A^{[l](i)}) + (1 - y^{(i)}) \log(1 - A^{[l](i)})) \tag{5}$$

After the cost is obtained, the back-propagation process is carried out. The formula is as follows:

$$dZ^{[l]} = dA^{[l]} \times R^{[l]'}(Z^{[l]}) \tag{6}$$

$$dw^{[l]} = \frac{1}{m} dZ^{[l]} \cdot A^{[l-1]T} \tag{7}$$

$$db^{[l]} = \frac{1}{m} \sum_{i=1}^{m} b^{[l](i)} \tag{8}$$

$$dA^{[l-1]} = w^{[l]T} \cdot dZ^{[l]} \tag{9}$$

After obtaining $dw^{[l]}$ and $b^{[l]}$, the learning rate is used to update the parameters, and then the above process is repeated to complete the training process of the model. Above, "·" is the vector internal product operation, "×" is the element by element multiplication operation.

4. Methodology

4.1. Loading the Pre-Training Model

Because of the small sample size of the ancient architecture dataset, in order to improve the model's training efficiency and model recognition accuracy, this paper adopts a model-based transfer learning technique, which means transferring and applying the Inception V3 model, trained on the ImageNet dataset, to the ancient architecture-type recognition task. In order to implement the model transfer, the Inception V3 pre-training model needs to be loaded first, namely, training the Inception V3 model on the ImageNet dataset to obtain a pre-training model with the model's parameters. Since the feature extraction methods of the data in ImageNet are similar but not identical to those of the ancient architecture dataset, in this paper, all layers in the Inception V3 model are added to the training process to achieve better recognition results.

4.2. Training Process of the Model

Figure 7 shows the training process of the improved Inception V3 model. The ancient architecture images in the training set are input into the pre-trained Inception V3 model, and the shallow features of the images are extracted via convolution in the first few layers of the model and pooled to reduce the feature map size, so as to further extract the deep features in the Inception block. The Inception block extracts feature maps of different sizes via parallel convolution layers; these feature maps are merged at the end of each Inception block and then input to the next layer. Then, all the extracted features are input into the global average pooling layer, and the dropout layer is added before the results of the global average pooling layer are input to the SoftMax classification layer, i.e., some neuron nodes are randomly ignored at a certain ratio (but the input layer and output layer are not changed). The SoftMax classification layer outputs the classification results and, thus, completes a forward propagation process. After the forward propagation is completed, the parameters in Inception V3 are updated according to the backpropagation algorithm (the ignored neuron nodes are not updated), thus completing an iterative process. The above steps are repeated until the end of the training process.

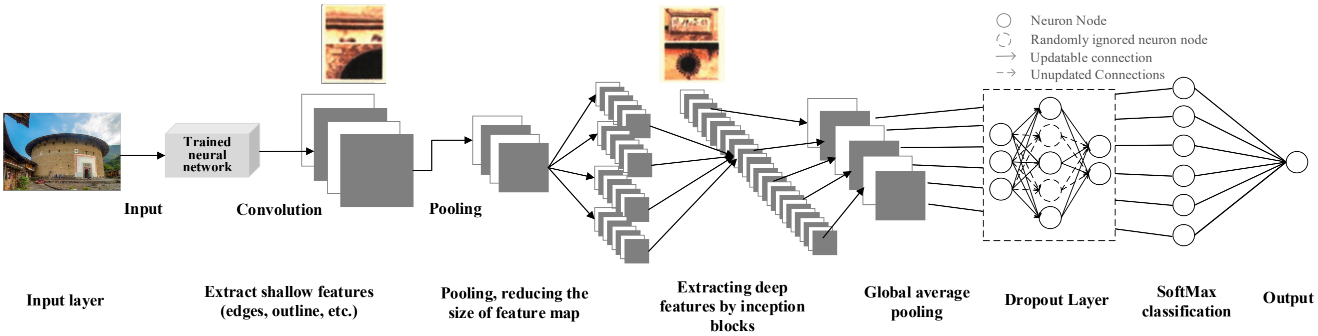


Figure 7. The training process of the improved Inception V3 model.

4.3. Settings of Model Hyperparameters

In the process of model training, the main hyperparameters, such as learning rate, epoch, batch size, optimizer, loss function, and evaluation metrics need to be optimized and tuned.

Learning rate: The learning rate determines how fast the objective function converges and has a substantial impact on the learning speed. When the learning rate is too large, the learning speed is fast, but the model parameters may not converge; if the learning rate is too small, the learning speed will be slow, which will lead to slow model convergence. After several tests, the empirical value of the learning rate is finally taken to be 0.001.

Epoch: This refers to the iteration times of the neural network. In this paper, the number of iterations is set to 20.

Batch size: This refers to the number of images input into the neural network at one time during training. If the batch size is too large, the memory will be overloaded; if the batch size is too small, the computing resources cannot be fully utilized, resulting in more waste. After testing, it is most appropriate to set the batch size to 16.

Optimizer: Since the Adam optimizer has the advantages of fast convergence and easy parameter tuning, as well as the ability to automatically adjust the learning rate, Adam is utilized as the optimizer for updating the parameters of the model in this paper. The update formula for the Adam parameters is as follows:

$$w_{t+1} = w_t - \frac{\alpha \cdot m_t}{\sqrt{v_t}} = w_t - \frac{\alpha \cdot (\beta \cdot m_{t-1} + (1-\beta_1) \cdot g_t)}{\sqrt{\beta_1 \cdot V_{t-1} + (1-\beta_2) \cdot g_t^2}} \tag{10}$$

where β_1 and β_2 are two hyper-parameters; β_1 controls the first-order momentum and β_2 controls the second-order momentum. The initial values of β_1 and β_2 are 0.9 and 0.999, respectively. At the same time, Adam can adaptively optimize the learning rate, taking the decay rate epsilon = 1×10^{-7}.

Loss function: For multi-category neural network models, the cross-entropy loss function is usually applied, and its formula is shown in Formula (11):

$$C = -\frac{1}{n}\sum_x [ylna + (1-y)\ln(1-a)] \tag{11}$$

where x represents the sample, y represents the real label, a represents the predicted output, and n represents the total number of samples.

Metric evaluation criteria: Since the task of this paper is to identify 6 types of ancient buildings, the traditional top 1 and top 5 accuracies cannot objectively measure the model's recognition effect. Therefore, the 1st and 5th accuracy rates are used as model evaluation criteria in this paper:

(1) 1st Accuracy: This represents the maximum value of top1 accuracy during 20 rounds of model training;

(2) 5th Accuracy: This represents the 5th largest value of top1 accuracy during 20 rounds of model training.

5. Experimental Verification and Result Analysis

The experiment environments are as follows:

(1) Experiment hardware environments: Intel(R) Xeon(R) Gold 5120 CPU @ 2.20GHz × 32 cores, NVIDIA Tesla T4 GPU, 1005 MHZ core frequency, 10,000 MHZ memory frequency, and a total of four video memories with each 15 GB of storage.

(2) Experiment software environments: Linux x86_64 operating system, driver version: 440.118.02.

(3) Network training environments: CUDA10.2, Python3.8.5, Tensorflow-gpu2.2.0.

5.1. Comparison and Analysis of Different Pre-Processing Methods

Data pre-processing methods have an important impact on the model's training results, and this section will compare and analyze the effects of different data pre-processing methods. Among them, the model training results after directly modifying the size without considering the aspect ratio of the image are shown in Figure 8, the model training results after center cropping according to the short side are shown in Figure 9, and the model training results after filling according to the long side are shown in Figure 10.

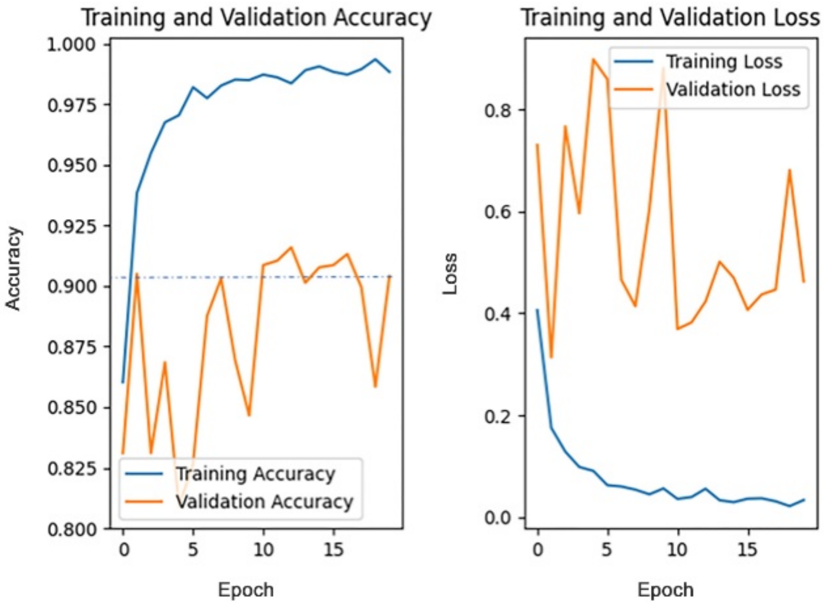


Figure 8. The training process of the pre-processing method of directly modifying the image size.

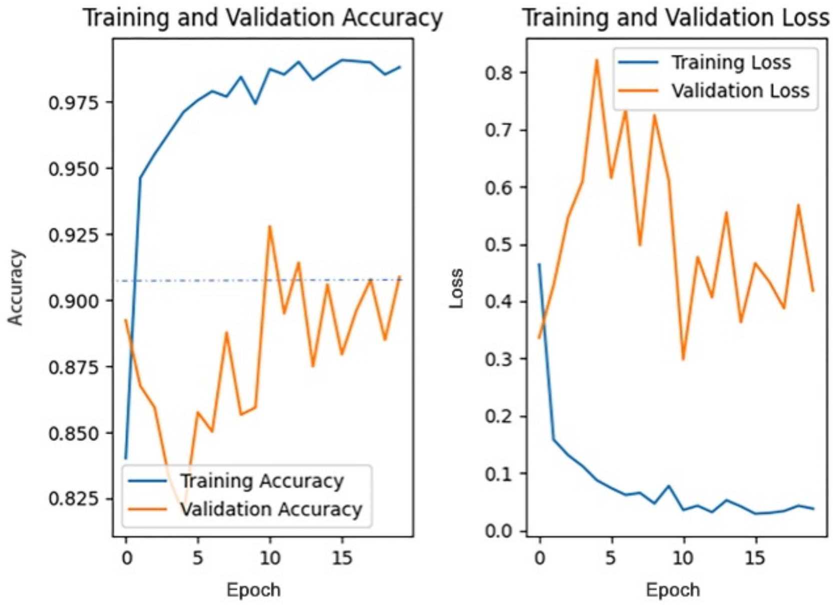


Figure 9. The training process of the pre-processing method of center-cropping.

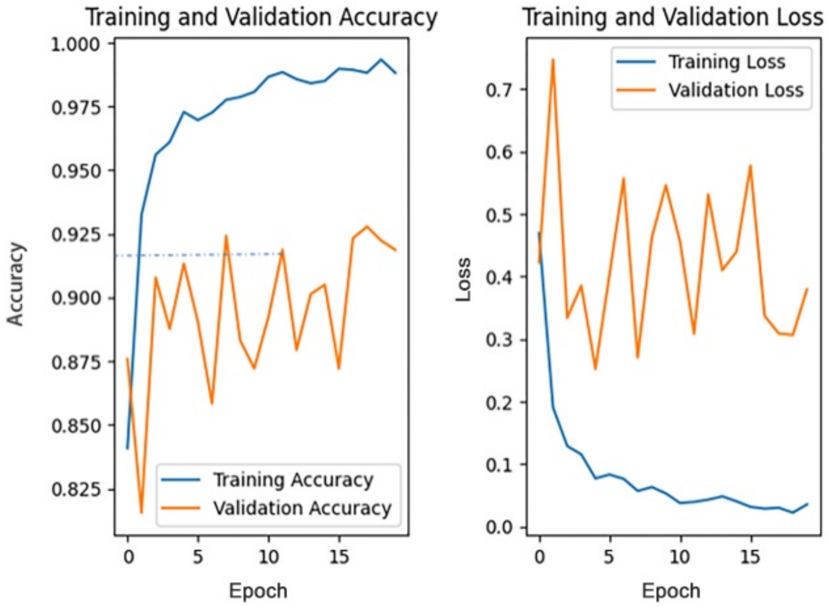


Figure 10. The training process of the pre-processing method of filling.

The 1st accuracy and 5th accuracy obtained by training the models according to three preprocessing methods are shown in Table 4.

Table 4. Comparison of the results of three pre-processing methods.

	1st Accuracy	5th Accuracy
directly modifying the size	92.16%	90.86%
cropping	92.65%	91.06%
filling	**92.68%**	**92.20%**

Because the 1st accuracy is accidental, in order to more objectively measure the model training results of each pre-processing method, this paper takes both the 1st accuracy and the 5th accuracy as the evaluation indicators of the model training effect. According to the above results, it can be observed that the 1st and 5th accuracies of the model obtained by the filling pre-processing method are higher than the other two pre-processing methods. Therefore, this paper finally chooses filling as the data pre-processing method for the ancient architecture recognition task. The most likely reasons for the above results are: (1) the pre-processing method of directly modifying the image size does not take into account the aspect ratio of the image, and some images may thus be distorted, which leads to the distortion of the feature map obtained by convolution, thus affecting the accuracy of the model; (2) although the pre-processing method of center cropping will not compress the image and cause distortion, for some specific buildings, their features may just be located at the edge of the image, and these features may just be clipped off, thus affecting the model recognition effect; (3) the pre-processing method of filling will not cause image distortion, nor will it lose some features due to clipping. Therefore, the filling pre-processing method is superior to the former two methods in both performance and results.

5.2. Comparison and Analysis of the Different Dropout Rates

This section discusses the impact of different dropout rates on the model training effect under the data pre-processing mode, based on filling, while other parameters remain unchanged. The dropout rate is generally no more than 0.5. This paper trains the model when the dropout rate is 0, 0.1, 0.2, 0.3, 0.4, and 0.5. The training process of the different dropout rates is shown in Figures 11–16.

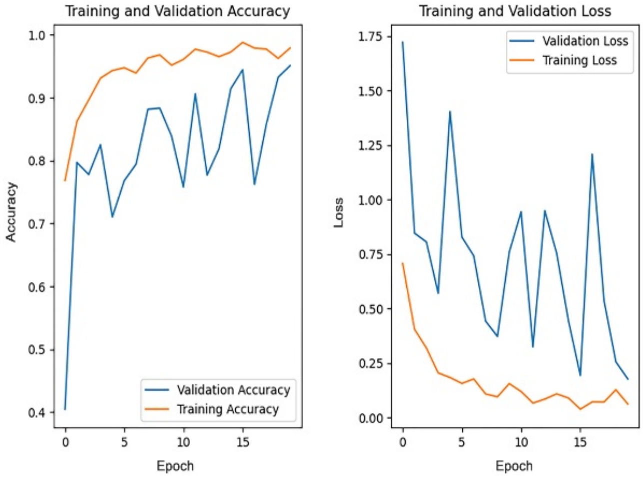


Figure 11. The training process with a dropout rate of 0.

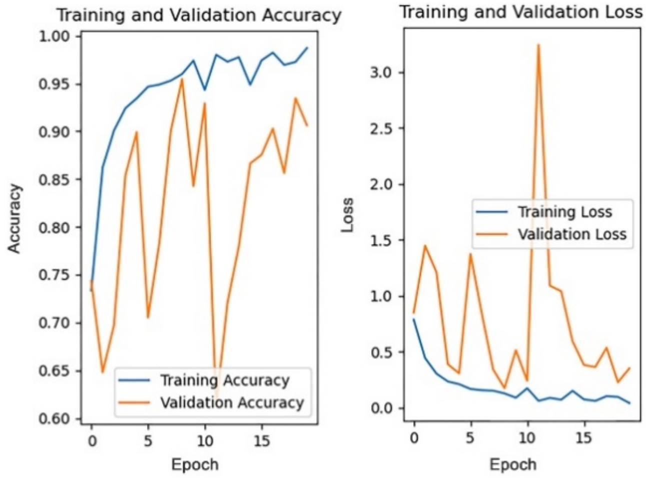


Figure 12. The training process with a dropout rate of 0.1.

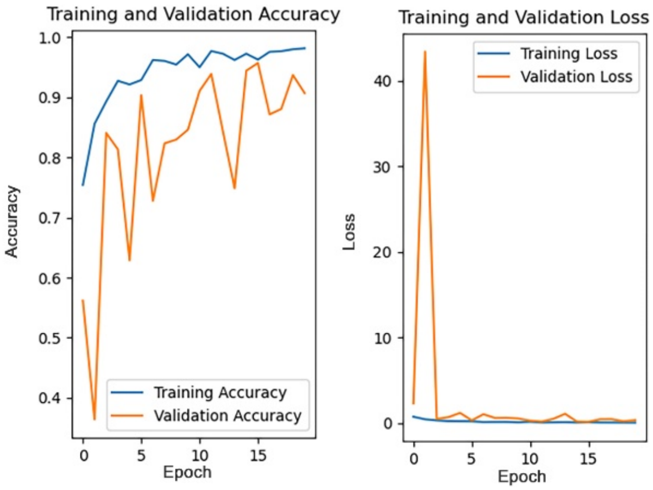


Figure 13. The training process with a dropout rate of 0.2.

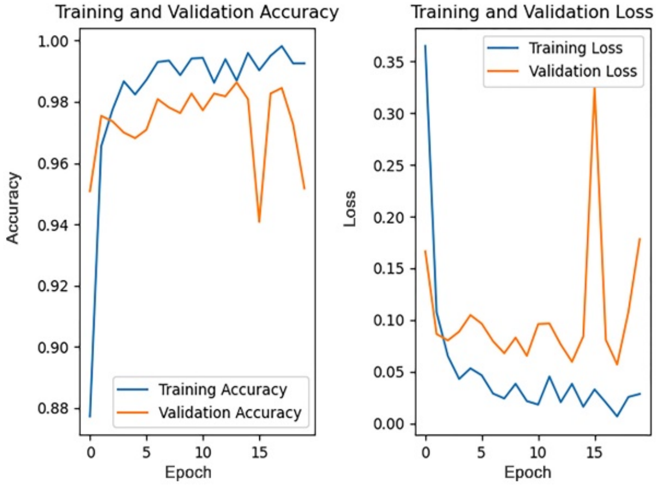


Figure 14. The training process with a dropout rate of 0.3.

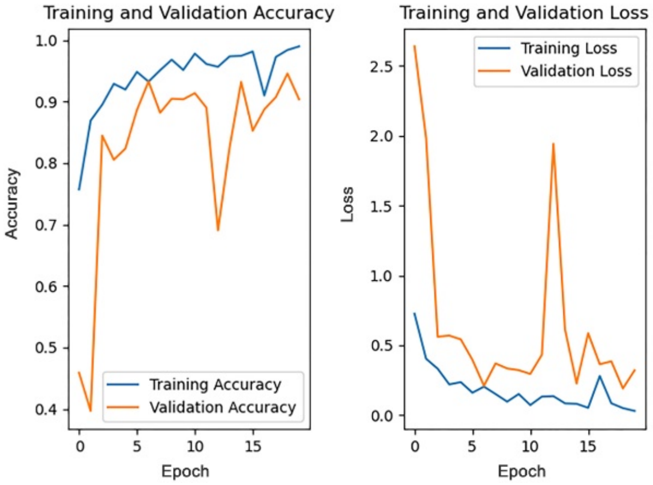


Figure 15. The training process with a dropout rate of 0.4.

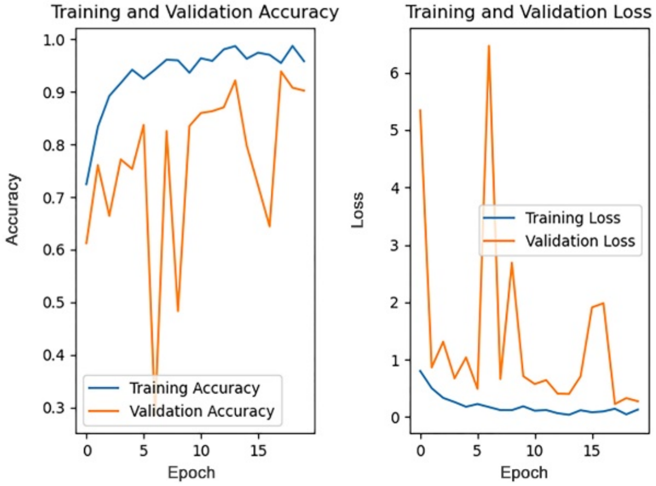


Figure 16. The training process with a dropout rate of 0.5.

Since the 1st accuracy value may have some deviation and contingency, likewise, the 1st accuracy and the 5th accuracy are also selected as the evaluation criteria. Table 5 shows the 1st and 5th accuracies under different dropout rates.

Table 5. The 1st/5th accuracy under different dropout rates.

Dropout Rate	1st Accuracy	5th Accuracy
0	95.09%	90.63%
0.1	95.45%	90.26%
0.2	95.72%	91.08%
0.3	**98.64%**	**98.27%**
0.4	94.54%	90.72%
0.5	93.90%	87.08%

From the above results, it can be seen that the recognition accuracy is the highest when the dropout rate is 0.3. In general, the dropout rate, as a hyper-parameter, is mainly used for regularization and there is no definite standard for its value. According to cross-validation experience, the effect is best when the dropout rate is 0.5. However, in this paper, the recognition accuracy is highest when the dropout rate is 0.3. One possible explanation is that buildings are usually tall and complex and contain more details. Therefore, the model will extract numerous features in the training process. For an object composed of a large number of features, when the dropout rate is too high, the model may ignore some important features in the training process, thus affecting the accuracy of the model recognition results. Therefore, when the dropout rate is 0.3, the accuracy is instead higher than that when the dropout rate is 0.5.

5.3. Comparison and Analysis of the Training Efficiency of Different Network Models

To verify the performance of the improved Inception V3 model proposed in this paper when on the ancient architecture recognition task, this paper finally chooses three classic convolutional neural networks, i.e., VGGNet19 [12], ResNet50 [13], and ResNet152 [19], to conduct a comparative analysis with the improved Inception V3 model.

The pre-trained weights on the ImageNet dataset are loaded into the four models, respectively, with each model trained for 20 rounds, then the accuracy of each model on the test dataset and the average training time are compared. Figures 17 and 18, respectively, show the changes in accuracy and loss during the training process of four models.

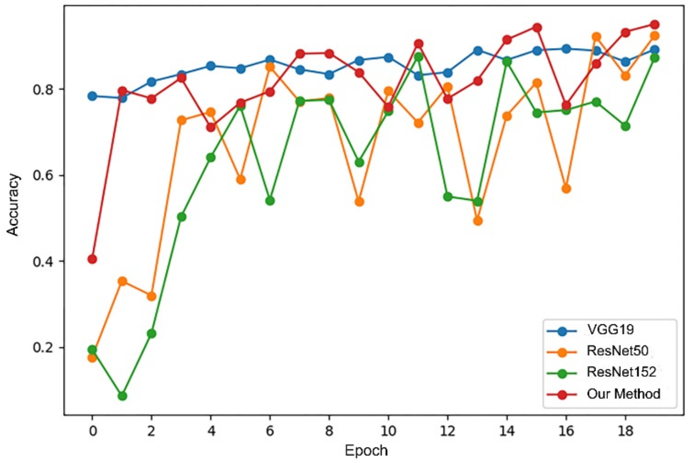


Figure 17. Accuracy of the verification set of four models.

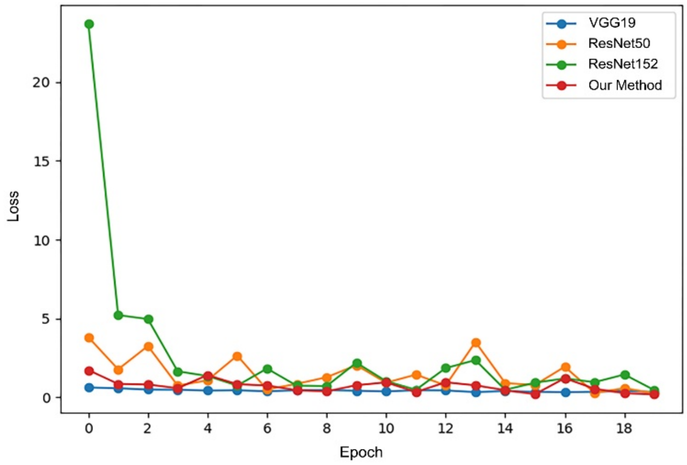


Figure 18. Loss change process of the validation set of four models.

Similarly, we use the 1st accuracy and the 5th accuracy as the evaluation criteria. The 1st accuracy and the 5th accuracy on the test dataset and the average training time of the four models are shown in Figure 19 and Table 6, respectively.

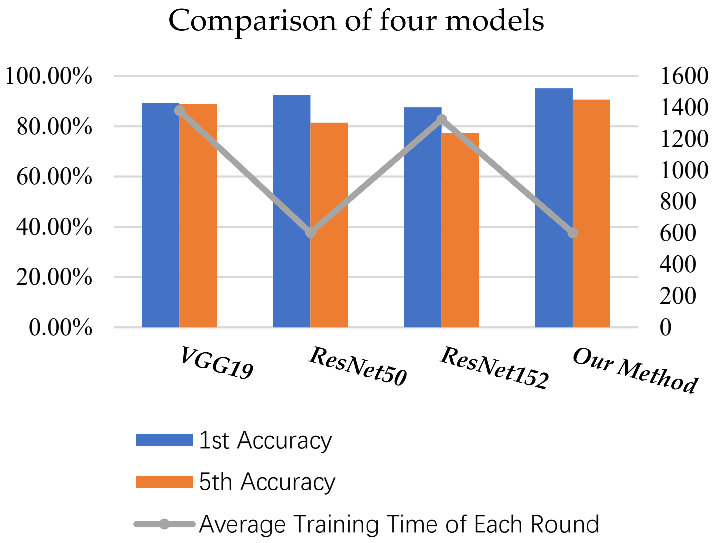


Figure 19. Comparison in terms of accuracy of the four models.

Table 6. Accuracy and average training time of the four models.

Model	1st Accuracy	5th Accuracy	Average Training Time
VGG19	89.35%	88.90%	1380.50 s
ResNet50	92.45%	81.44%	602.85 s
ResNet152	87.53%	77.25%	1323.60 s
Our Method	**95.09%**	**90.63%**	**601.55 s**

It can be seen from the above results that the 1st and 5th accuracies of the improved Inception V3 both show the highest values, while the average training time is also the shortest with the same hardware resources, showing that it performs the best, comprehensively. The cause for this lies in the fact that the last layers of the VGGNet19, ResNet50, and ResNet152 networks use the full connection layer, inevitably increasing the network training parameter scale. In comparison, there is no full connection layer in the improved Inception V3 model, and it has a higher training speed. In addition, because the ResNet152 model contains 152 layers and has the deepest structure, it is more prone to overfitting, and the training time is longer. To sum up, compared with the discussed models, the improved Inception V3 model adopted in this paper has the best results when performing the ancient architecture recognition task.

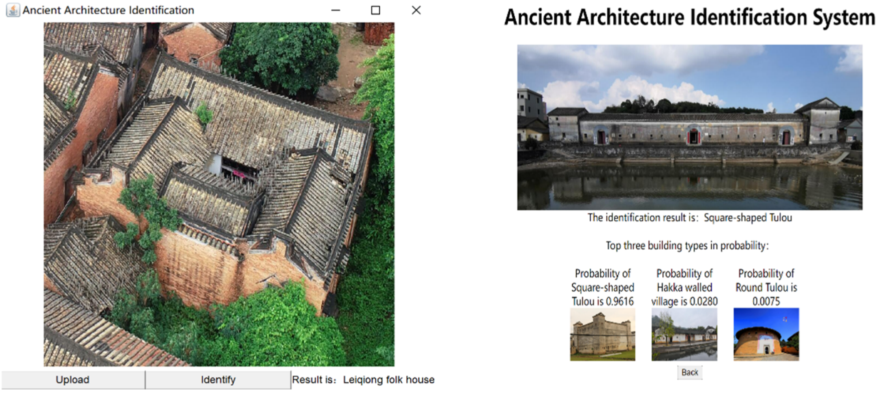
5.4. Realization of Recognition System

In this paper, we adopt SoftMax as the activation function, formulated as follows:

$$y_i = \frac{e^{x_i}}{\sum_{i=1}^{n} e^{x_i}} \tag{12}$$

where x_i is the score given when one image belongs to type i after feature extraction and weight addition, while y_i is the probability that the image belongs to type i. After being processed by the SoftMax activation function, the scores given when images belong to each type are converted to the corresponding probabilities, and the type with the highest probability value is the most likely type to which an ancient building belongs. In order to eliminate the interference of other images, it is only when the probability value of the highest probability type is greater than 60% that it can be determined that the building belongs to the corresponding category; otherwise, it will be included in other categories, namely, types other than the square-shaped Tulou, Canton ancestral hall, Leiqiong folk house, San-Jian Liang-lang folk house, Hakka walled village, and round Tulou.

Based on the identification model in this paper, our team has developed the B/S architecture and C/S architecture ancient building recognition system, which improved the work efficiency of the research team and the professionals. The running system is shown in Figure 20, where left subfigure and right subfigure represent the recognition results under the C/S architecture and B/S architecture respectively.



Result is: Leiqiong folk house

Ancient Architecture Identification System

The identification result is: Square-shaped Tulou

Top three building types in probability:

Probability of Square-shaped Tulou is 0.9616

Probability of Hakka walled village is 0.0280

Probability of Round Tulou is 0.0075

Figure 20. Display of the working ancient architecture identification system.

6. Discussion

The automatic identification of ancient building types has important research significance and application value for studying the development of society and cultural civilization and is also the application of an asymmetric system. In order to improve the accuracy of the machine vision recognition of ancient buildings, based on the convolutional neural network model, this paper proposes an improved Inception V3 model for ancient building recognition. The automatic recognition model proposed in this paper is mainly improved in the following aspects: (1) a dropout layer is added after the global average pooling layer of the Inception V3 model, which effectively alleviates the overfitting problem in the model learning process; (2) on the one hand, the integration of transfer learning and the ImageNet dataset into the training process of the model speeds up the training speed of the model, while on the other hand, it can further alleviate the overfitting problem caused by insufficient training due to a small sample size; (3) this paper also optimizes the network super-parameters of the recognition model through ablation experiments: the recognition effect of the model is the best when the data preprocessing method is set to the filling mode, and the dropout rate is set to 0.3 at this time. The data set of ancient buildings, independently constructed by the team of the South China University of Technology, is taken as the experimental data; then, the recognition effects of several convolutional network models, i.e., VGGNet19, ResNet50, and ResNet152, are compared and analyzed. The experimental results show that the accuracy of the improved Inception V3 model, proposed in this paper to identify ancient buildings, reaches 98.64% at the highest level. Compared with other classical models, the recognition model proposed in this paper has the best recognition accuracy and the least model training time, which validates the effectiveness of promoting performance in the ancient building recognition task of our proposed method.

The research in this paper provides a new idea and method for the recognition of ancient buildings, based on convolution. In future research, we will further carry out the recognition of ancient building materials and other features and will further improve the recognition effect by combining data enhancement, target recognition, and other methods.

Author Contributions: Conceptualization, X.W. and J.T.; methodology, X.W. and J.L.; software, J.L., Z.Z. and Z.D.; validation, X.W., J.L., L.W. and Z.Z.; formal analysis, X.W.; investigation, Z.D.; resources, X.W. and J.T.; data curation, X.W.; writing—original draft preparation, J.L., Z.D. and Z.Z.; writing—review and editing, X.W. and X.Z.; visualization, J.L. and Z.Z.; supervision, X.W. and W.B.; project administration, X.W.; funding acquisition, X.W., J.T. and C.M. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Fundamental Research Funds for the Central Universities (Nos. BLX202140 and BLX201923), Guangdong Basic and Applied Basic Research Foundation (No. 2021B1515020013), and Guangdong Provincial Philosophy and Social Science Planning Project (No. GD21CYS05).

Data Availability Statement: Not applicable.

Acknowledgments: We would like to show our sincere gratitude to everyone who has provided relevant suggestions and offered timely help in writing this paper.

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

References

1. Di, H. Talking about digital technology in the protection of ancient buildings. *Identif. Apprec. Cult. Relics* **2018**, *3*, 130–132.
2. Yanni, W.; Huiqin, W.; Jianping, W. A building recognition algorithm based on local feature and shape contour matching. *J. Xi'an Univ. Archit. Technol. (Nat. Sci. Ed.)* **2017**, *49*, 752–756.
3. Wu, Y. Classification of Ancient Buddhist Architecture in Multi-Cultural Context Based on Local Feature Learning. *Mobile Inf. Syst.* **2022**. [CrossRef]
4. Hasan, M.; Kabir, S.R.; Akhtaruzzaman, M.; Sadeq, M.J.; Alam, M.M.; Allayear, S.M.; Uddin, M.; Rahman, M.; Forhat, R.; Haque, R.; et al. Identification of construction era for Indian subcontinent ancient and heritage buildings by using deep learning. In Proceedings of the International Congress on Information and Communication Technology, London, UK, 20–21 February 2020; pp. 631–640.

5. Zhang, S.; Chen, S.; Zhang, J.; Cai, Z.; Hu, L. Image annotation of ancient Chinese architecture based on visual attention mechanism and GCN. *Multimed. Tools Appl.* **2022**, *81*, 39963–39980. [CrossRef]
6. Yang, S.; Shengyang, L.; Shao, Y.; Zheng, H. Building recognition method based on improved HOG feature. *Comput. Eng. Appl.* **2018**, *54*, 196–200.
7. Freeman, W.T.; Adelson, E.H. The design and use of steerable filters. *IEEE Trans. Pattern Anal. Mach. Intell.* **1991**, *13*, 891–906. [CrossRef]
8. Yu, F.; Koltun, V.; Funkhouser, T. Dilated residual networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 472–480.
9. Chollet, F. Xception: Deep learning with depthwise separable convolution. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 1251–1258.
10. Lecun, Y.; Bottou, L.; Bengio, Y.; Haffner, P. Gradient-based learning applied to document recognition. *Proc. IEEE* **1998**, *86*, 2278–2324. [CrossRef]
11. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. ImageNet classification with deep convolutional neural networks. *Commun. ACM* **2017**, *60*, 84–90. [CrossRef]
12. Simonyan, K.; Zisserman, A. Very deep convolutional networks for large-scale image recognition. *arXiv* **2014**, arXiv:1409.1556.
13. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.
14. Szegedy, C.; Vanhoucke, V.; Ioffe, S.; Shlens, J.; Wojna, Z. Rethinking the inception architecture for computer vision. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 2818–2826.
15. Wu, Z.; He, S. Improvement of the AlexNet Networks for Large-Scale Recognition Applications. *Iran. J. Sci. Technol. Trans. Electr. Eng.* **2020**, *45*, 493–503. [CrossRef]
16. Sengupta, A.; Ye, Y.; Wang, R.; Liu, C.; Roy, K. Going deeper in spiking neural networks: VGG and residual architectures. *Front. Neurosci.* **2019**, *13*, 95. [CrossRef] [PubMed]
17. Younis, A.; Qiang, L.; Nyatega, C.O.; Adamu, M.J.; Kawuwa, H.B. Brain Tumor Analysis Using Deep Learning and VGG-16 Ensembling Learning Approaches. *Appl. Sci.* **2022**, *12*, 7282. [CrossRef]
18. Ding, X.; Zhang, X.; Han, J.; Ding, G. Scaling up your kernels to 31×31: Revisiting large kernel design in cnns. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 19–24 June 2022; pp. 11963–11975.
19. Qiaohua, Q.; Dawei, X.; Mingnan, L.; Jin, T. A study on the classification of traditional village images based on convolutional neural network. *City Plan. Rev.* **2020**, *44*, 52–58.
20. Kun, G. Research on Classification of Architectural Style Image Based on Convolution Neural Network. Master's Thesis, Wuhan University of Technology, Wuhan, China, 2017.
21. Xingyi, W. Research on digital protection of ancient buildings in the era of information technology. *Creat. Living* **2021**, *7*, 136–137.
22. Yu, F.; Xiu, X.; Li, Y. A Survey on Deep Transfer Learning and Beyond. *Mathematics* **2022**, *10*, 3619. [CrossRef]
23. Zehong, W.; Houquan, L. Building Recognition Based on Transfer Learning and Adaptive Feature Fusion. *Comput. Technol. Dev.* **2019**, *29*, 40–43.
24. 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.
25. Ioffe, S.; Szegedy, C. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In Proceedings of the 32nd International Conference on Machine Learning, Lille, France, 6–11 July 2015; pp. 448–456.
26. Srivastava, N.; Hinton, G.; Krizhevsky, A.; Sutskever, I.; Salakhutdinov, R. Dropout: A simple way to prevent neural networks from overfitting. *J. Mach. Learn. Res.* **2014**, *15*, 1929–1958.
27. Hinton, G.E.; Srivastava, N.; Krizhevsky, A.; Sutskever, I.; Salakhutdinov, R.R. Improving neural networks by preventing co-adaptation of feature detectors. *arXiv* **2012**, arXiv:1207.0580.