remote sensing

MDPI

Article

Interoperability Study of Data Preprocessing for Deep Learning and High-Resolution Aerial Photographs for Forest and Vegetation Type Identification

Feng-Cheng Lin [1] and Yung-Chung Chuang [2,*]

[1] Department of Information Engineering and Computer Science, Feng Chia University, Taichung 407802, Taiwan; fclin@fcu.edu.tw
[2] Department of Urban Planning and Spatial Information, Feng Chia University, Taichung 407802, Taiwan
* Correspondence: yungcchuang@fcu.edu.tw; Tel.: +886-4-24517250 (ext. 3371); Fax: +886-4-24517278

Abstract: When original aerial photographs are combined with deep learning to classify forest vegetation cover, these photographs are often hindered by the interlaced composition of complex backgrounds and vegetation types as well as the influence of different deep learning calculation processes, resulting in unpredictable training and test results. The purpose of this research is to evaluate (1) data preprocessing, (2) the number of classification targets, and (3) convolutional neural network (CNN) approaches combined with deep learning's effects on high-resolution aerial photographs to identify forest and vegetation types. Data preprocessing is mainly composed of principal component analysis and content simplification (noise elimination). The number of classification targets is divided into 14 types of forest vegetation that are more complex and difficult to distinguish and seven types of forest vegetation that are simpler. We used CNN approaches to compare three CNN architectures: VGG19, ResNet50, and SegNet. This study found that the models had the best execution efficiency and classification accuracy after data preprocessing using principal component analysis. However, an increase in the number of classification targets significantly reduced the classification accuracy. The algorithm analysis showed that VGG19 achieved the best classification accuracy, but SegNet achieved the best performance and overall stability of relative convergence. This proves that data preprocessing helps identify forest and plant categories in aerial photographs with complex backgrounds. If combined with the appropriate CNN algorithm, these architectures will have great potential to replace high-cost on-site forestland surveys. At the end of this study, a user-friendly classification system for practical application is proposed, and its testing showed good results.

Keywords: data preprocessing; deep learning; high-resolution aerial photographs; forest and vegetation type identification

check for updates

Citation: Lin, F.-C.; Chuang, Y.-C. Interoperability Study of Data Preprocessing for Deep Learning and High-Resolution Aerial Photographs for Forest and Vegetation Type Identification. *Remote Sens.* **2021**, *13*, 4036. https://doi.org/10.3390/rs13204036

Academic Editors: Chiman Kwan and Peng Fu

Received: 11 July 2021
Accepted: 26 September 2021
Published: 9 October 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

High-resolution aerial photographs can provide considerable surface and terrain information and can be used to interpret forest types. Previous studies and applications have focused on the interpretation and analysis of large-scale surface conditions using different spectral and spatial resolution photographs or images (e.g., SPOT, IKONOS, FORMOSAT-2, QuickBird, and World-View2) [1–5]. New types of remote sensing platforms such as ultra-high-resolution satellites or unmanned aerial vehicles (UAVs) can observe and analyze plant canopies by adding spatial details. Synthetic aperture radar (SAR) and ground or airborne laser scanning can capture the three-dimensional structure of forest vegetation at all angles. In addition, data sharing and open information have also shown rapid growth. These ever-increasing types of remote sensing have encountered many challenges in their application to forest vegetation surveys. These challenges include the increasing amount of data, computational load, and complexity of targets with a continuous increase

in dimensions (space, time, spectrum) in the data structure. Aerial photographs have higher resolution than satellite images. Therefore, they can be used to extract detailed tree growth information [6]. However, informational analysis is limited by the variable scale and spectral resolutions of the images themselves. This leads to decreased effectiveness in identifying land cover types using manual digitizing on geographic information system (GIS) platforms. Therefore, digital image processing and algorithm-based image classification have been used to solve these problems.

In the past, data analysis methods related to remote sensing detection often used feature engineering to perform conversion operations from input data and manually create latent variables before modeling. Examples in the field of vegetation remote sensing are band combination, data augmentation, and texture measurement. These processing methods add value to image data through calculation processing or the mathematical conversion of original images. For example, the contrast of various objects in an image can be increased through image enhancement processes to improve the classification rate (or interpretation rate); additionally, images that are blurred or unclear can be repaired using image restoration technologies [7,8]. In image interpretation conducted using traditional machine learning, mathematical transformation is usually employed and involves specific algorithms that analyze features such as agriculture, medical images, land cover, and outdoor scenes [9–12]. The most common feature engineering method applied to remote sensing imagery is the gray level co-occurrence matrix (GLCM) method, which provides information on the specific spatial distribution of grayscale values in an image. However, this information must be quantified into a single form to obtain the image layer for subsequent analysis. Haralick [13] suggested a variety of texture statistics that can be used to quantify the GLCM, such as contrast, correlation, dissimilarity, energy, entropy, homogeneity, and variance. Several studies have used the GLCM to improve the feasibility of image interpretation and classification [2,14–16].

The above data processing methods have their advantages, but they are often limited by the excessive amount of data; as a result, the most effective predictors are usually unable to derive data and have low efficiency. In addition, the most suitable predictor for plant classification may experience difficulties because it needs to understand how the physiological and structural characteristics of plants are related to interactions with the electromagnetic signals measured by the sensors. In contrast, through deep learning, the neural network itself can learn the data transformation that best solves the current problem.

Unlike machine learning, in which feature analysis is performed after repeated training through human operation, deep learning enables automatic feature extraction. Deep learning, which was founded by Schmidhuber [17], is a branch of machine learning in which multilayered neural networks are employed to perform linear or nonlinear data transformations and provide features that can be used to identify specific objects or images. This type of feature learning can replace manual identification and has relatively high efficiency for feature extraction.

A convolutional neural network (CNN) is a deep learning algorithm that has been widely used for image or signal recognition in recent years. CNNs were proposed by Hubel and Wiesel [18], who discovered that a unique network function can reduce the complexity of feedback neural networks. In a CNN, each input is connected to a receptive field, which is a small region in its previous layer that extracts local features. Each computing layer in the network contains several local feature maps. A feature projection result with displacement invariance can be obtained after being processed through the sigmoid function. Moreover, each convolution layer is followed by a calculation layer for local average and secondary extraction, which reduces the likelihood of excessive signal divergence. This special algorithmic structure of local weight sharing has unique advantages in computer image processing; thus, scholars have applied remote sensing image processing in recent years [19–21].

For CNN optimization, Hinton and Salakhutdinov [22] found that multilayer neural networks can be better trained by conducting layer-by-layer pretraining and that a lower

error rate can be obtained using these networks on the digital image data in the MNIST database than data obtained using the support vector machines algorithm. Glorot et al. [23] proposed a rectified linear unit (ReLU) startup function, which is one of the most widely used startup functions. Krizhevsky et al. [24] proposed AlexNet, an eight-layer deep neural network that combines the ReLU start function with dropout to prevent overfitting. Krizhevsky trained the network directly on two GTX 580 graphics processing units without using the layer-by-layer pretraining method. AlexNet secured first place in the ImageNet Large Scale Visual Recognition Competition (ILSVRC) 2012 and outperformed the second-place device by 10.9% in terms of the error rate. Various algorithm models have been proposed since the AlexNet model was developed, including VGG [25], GoogleNet [26], ResNet [27], and DenseNet [28]. Increasing the number of convolutional layers and transformations can extract higher-level feature functions and more abstract concepts, thereby revealing more complex and hierarchical relationships. A series of studies has shown that increasing the depth of CNNs can indeed enhance the retrieval of vegetation-related information in remote sensing data.

Some relevant studies, such as vegetation and land cover classification, are worthy of attention. Pixel-based and object-based methods can be used in land cover classification. Kwan et al. [29] observed that red-green-blue (RGB) and near infrared (NIR) bands can perform better than the NDVI and EVI approaches in vegetation detection based on land cover classification. For land cover classification, Kwan et al. [30] used a limited number of bands (RGB + NIR and RGB + NIR + LiDAR) and EMAP-based augmentation; moreover, they gave a comprehensive performance evaluation of two CNN-based deep learning (customized CNN and CNN-3D) algorithms. For vegetation detection, Ayhan et al. [31] proposed a novel object-based vegetation detection approach using NDVI, computer vision, and machine learning techniques. They also compared the performance of DeepLabV3+ (RGB and NDVI-GB bands) with a proposed CNN-based deep learning method (RGB and RGB-NIR bands) and NDVI-ML (RGB-NIR). Incidentally, Li and Stein [32] examined the use of graph convolutional networks (GCNs) to extract high-level structural features for land use mapping from high-resolution satellite images.

The above deep learning and CNN studies showed that a CNN model can reveal the precise spatial representations and object characteristics of data collected by a sensor. This advantage will simplify traditional remote sensing detection in which researchers must use complex reference data and plant physiology. However, reducing the consideration of electromagnetic signals results in a certain degree of gray-box or black-box work. Therefore, checking the characteristics of input data and model algorithms to find regularities in the CNN applied to the classification of plant traits and communities will be of great help to subsequent extended applications. On that basis, this study conducted a combined analysis of sensor data type, CNN algorithm, and target classification complexity and evaluated their practical application to large-area forest surveys.

The main purpose of the research was to:

1. Compare the effects of data preprocessing;
2. Compare the impact of the number of classification targets on the classification results;
3. Compare the impact of CNN methods and architecture on the combination of high-resolution aerial photographs and deep learning to identify forest and vegetation types.

Section 2 introduces the research area and research methods, Section 3 explains the research results, and Section 4 discusses the comparative analysis results in depth.

2. Materials and Methods

The main purpose of this research was to evaluate the feasibility of using high-resolution aerial photographs to classify forest vegetation and to compare the results of interactions between model operation variables to determine regularity. The study area and materials used in this research are summarized below.

2.1. Study Area and Workflow

This study selected forestland in a suburban area of Taipei, Taiwan as the study area (Figure 1). The total area of the forestland is approximately 2486 ha, and it is mostly located in a shallow mountainous area near neighboring cities. The gentle slopes reduce the possibility of shade or terrain shade while conducting aerial photography, and many different types of forests exist in the area.

The survey and research methods of this study were divided into five steps: (1) collection of processable high-resolution aerial photographs, (2) field surveys of the forestland, (3) image preprocessing, (4) deep learning methods for forestland classification and effectiveness evaluation, and (5) actual development of a forestland classification system. This study first collected orthorectified high-resolution aerial photographs (2016/6–2018/8) of the study area. Several permanent plots were selected for field surveys afterward (on-site investigation conducted from 2018/7–2018/8). The survey contents included forest types, vegetation species, and xyz coordinates. In the third stage, image processing of aerial photographs was carried out. We applied content simplification (CS) and principal component analysis (PCA) transformation to generate layers as new input data that could best highlight the two-dimensional texture and three-dimensional shape of different forest types and vegetation species. Automatic forestland classification testing was conducted in the final stage by different deep learning models (VGG19, ResNet50, and SegNet). Finally, the effectiveness of the input data and models was evaluated, the best solution was found, and a prototype system was proposed. Figure 2 displays the identification process used in this study.

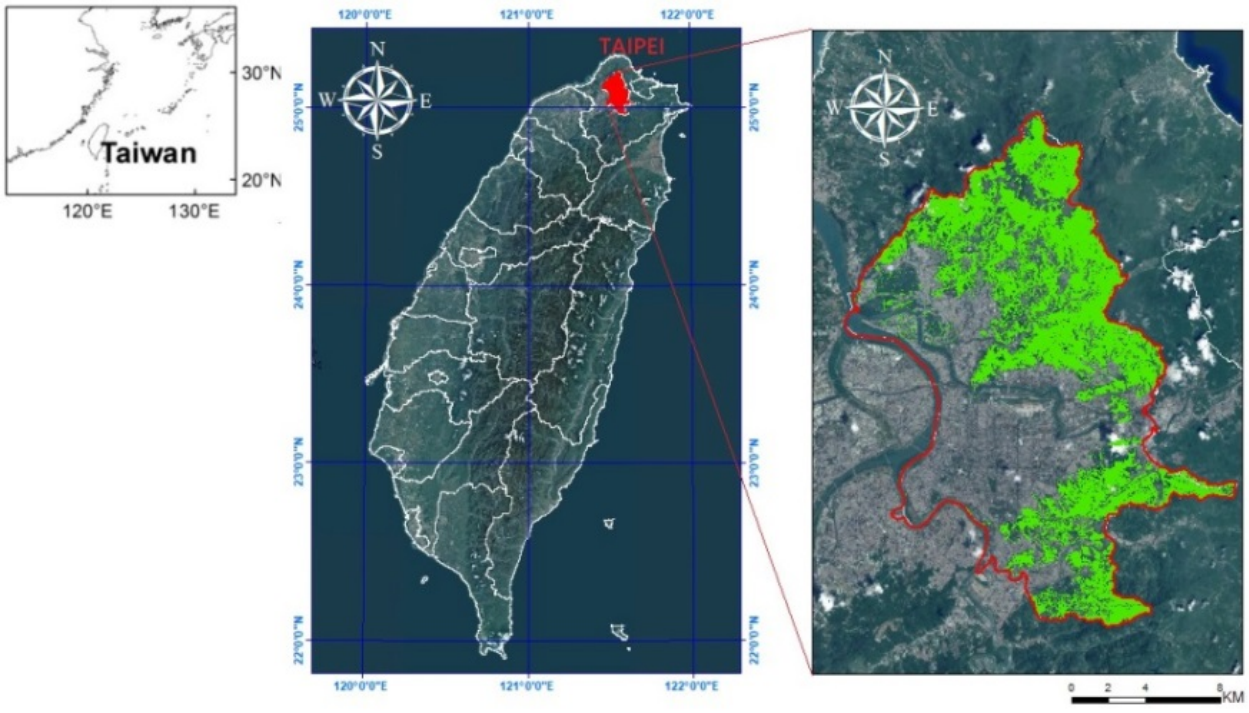


Figure 1. Location of the research area in Taipei, Taiwan.

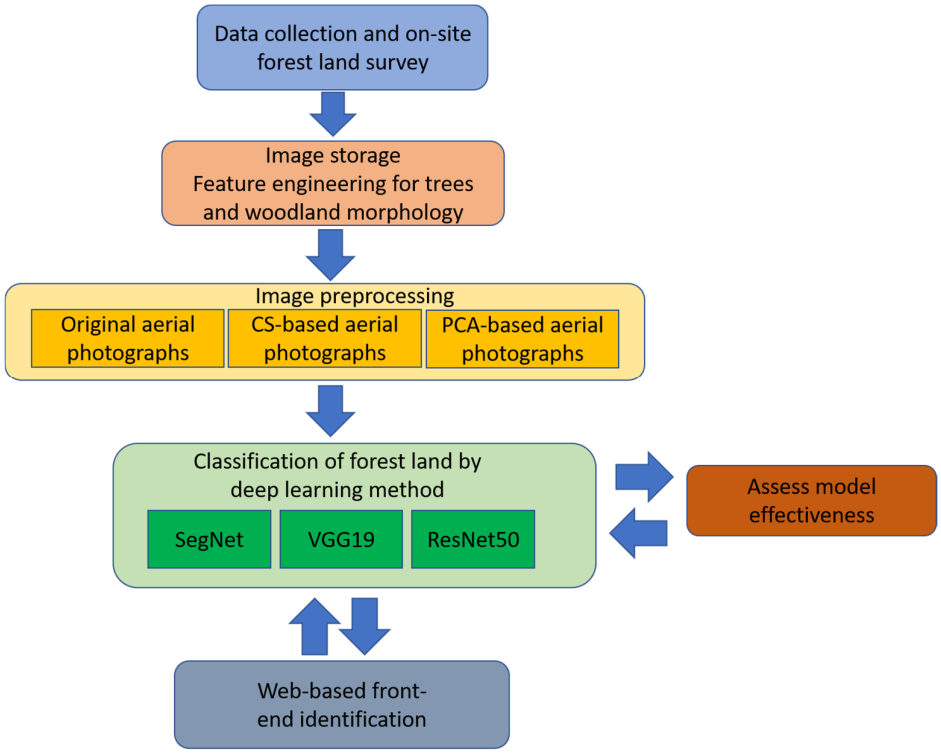


Figure 2. Flowchart representing the methodology used for forest and vegetation type identification.

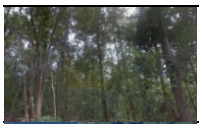
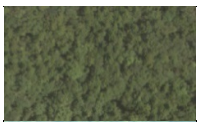
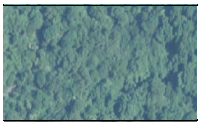
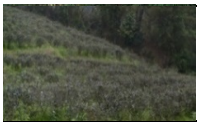
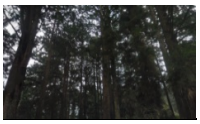
2.2. Forestland Investigation and Category Setting

This study investigated forest conditions in the study area. The investigation process involved pre-revision planning, on-site surveying, sample plot setting, and investigation (each plot size was 10 × 10 m^2). The contents of the forestland survey included flora and forest-type surveys. The flora survey was based on species and listed all the plant species surveyed in a certain area, detailing the species group, amount, structure, distribution, dynamics, and reproductive environment of each species. For the subsequent internal process, the survey results were categorized into representative clusters, and exclusive codes were set to the clusters so that the samples from the 14 types of forestland were classified accordingly. Each forest and vegetation type is presented in Table 1, and their names are as follows: (1) Artificial coniferous and broad-leaved mixed forest, (2) Natural coniferous and broad-leaved mixed forest, (3) Natural grasslands, (4) Natural broad-leaved mixed forest, (5) Dryland farming, (6) Artificial pine forest, (7) Citrus orchard, (8) *Cryptomeria japonica*, (9) Taiwan acacia, (10) Makino Bamboo, (11) Tea plantation, (12) *Liquidambar formosana*, (13) *Indocalamus oldhamii*, and (14) *Cunninghamia lanceolata*. Because some forest and vegetation types in our collected dataset were similar, we pooled these types into one group and labeled them with a group identification number (ID): {1, 3}, {11}, {8, 14, 6}, {10, 13}, {7, 5}, {2}, and {4, 9, 12}. We selected seven distinct forest and vegetation types from the collected dataset—those with IDs 1, 2, 4, 7, 8, 10, and 11—and used them as the seven types of forest and vegetation in the dataset.

Field surveys of forestland were conducted to confirm the spatial distribution and coordinate position of various forest and vegetation species. Eleven primary forest cover types were found in the protection forest. With aerial photography, we also found tea plantations, citrus orchards, and dryland farming, which are all illegal uses of the land. This study thus focused on all 14 types of forest cover and partial sample sizes (256 × 256). In the image classification process, the aerial photographs (spatial resolution: 0.46–0.61 m) with the on-the-ground data layer produced through the field survey allowed us to extract all 14 types of forest cover based on brightness, image clarity, and cloud coverage in 33 aerial photographs. These photographs were divided into 2541 small images (256 × 256 pixels).

Here, 2041 images were used for model training, and the remaining 500 images were used for test verification (each image was repeated 30 times). The 500 simulation results of the model corresponded to the data on the actual locations and indicated whether the model interpretation was correct.

Table 1. Fourteen types of forest and vegetation.

Forest and Vegetation Type	Ground Truth	Aerial Photograph	Forest and Vegetation Type	Ground Truth	Aerial Photograph
(1)			(8)		
(2)			(9)		
(3)			(10)		
(4)			(11)		
(5)			(12)		
(6)			(13)		
(7)			(14)		

2.3. Image Preprocessing for CS and PCA

Various forest types and vegetation species in the original aerial photographs were very similar in terms of color, texture, shape, and spatial pattern. It was thus difficult to identify their types directly. Since the key to the operation of deep learning models lies in how effectively they extract information from spatial data and perform learning, providing appropriate spatial images before the model is trained is the key to the success of learning. However, additional feature engineering, such as conversion, vegetation index, preprocessing, and speckle reduction, may introduce information loss and reduce model accuracy [6,33,34]. Therefore, it is very important to filter the features that are helpful to the classification in the process of data conversion.

In previous photo-interpretation studies using artificial interpretation of tree species, some crown typologies and definitions were used as the basis for classification criteria, such as "crown size", "crown status", "crown contour", "crown architecture", "foliage cover", "foliage texture", "color", and "phenology" [35]. However, its definition is too detailed and may not be suitable for high-altitude aerial photographs. The feature extraction process

of image conversion is too complicated, and the operation process is also too difficult for on-site forestry practitioners. Therefore, this research sought to propose two relatively simple data pre-processing methods for forestry practitioners (which can be implemented using general GIS software)

First, we introduced content simplification and simplified the classification criteria into four indicators: "the light surface of the tree canopy", "the shadow surface of the tree canopy", "gaps between trees", and "bare ground". Generally speaking, coniferous trees have smaller crowns, so the figure area of "the light surface of the tree canopy" and "the shadow surface of the tree canopy" is smaller than that of broad-leaved trees. The spacing between trees in natural forests is mostly smaller than that in artificial forests, so the number and area of "gaps between trees" will be lower. The appearance of the layer converted using only these four classification criteria seemed rough and had little information. However, for different forest types, the block size, arrangement, and texture characteristics of the four classification criteria were clearly separated.

In the CS preprocessing process, researchers can use commonly used software such as ArcGIS or ENVI. In the process of ISODATA cluster method (e.g., unsupervised classification function in ArcGIS 10.1 geographic information software), regardless of the initial set number of classifications, this study used the "reclassify function" to divide its simplified classification into four categories: (1) darker black: "gaps between trees"; (2) whiter: "bare ground"; (3) bright green: "the light surface of the tree canopy"; (4) dark green: "the shadow surface of the tree canopy". Compared with the noise problem faced when using high-resolution images for pixel-based classification, this setting can achieve an effect similar to segmentation for object-based image analysis (Table 7). Although the spectral distribution is narrower than the original image (Figure 3), the spatial distribution of the image can clearly show the tree shape and the positional relationship between the trees. Finally, by using this method, specific characteristics of different forest types and vegetation species were strengthened, and noise that may have influenced the input data was eliminated to improve the classification ability of deep learning algorithms.

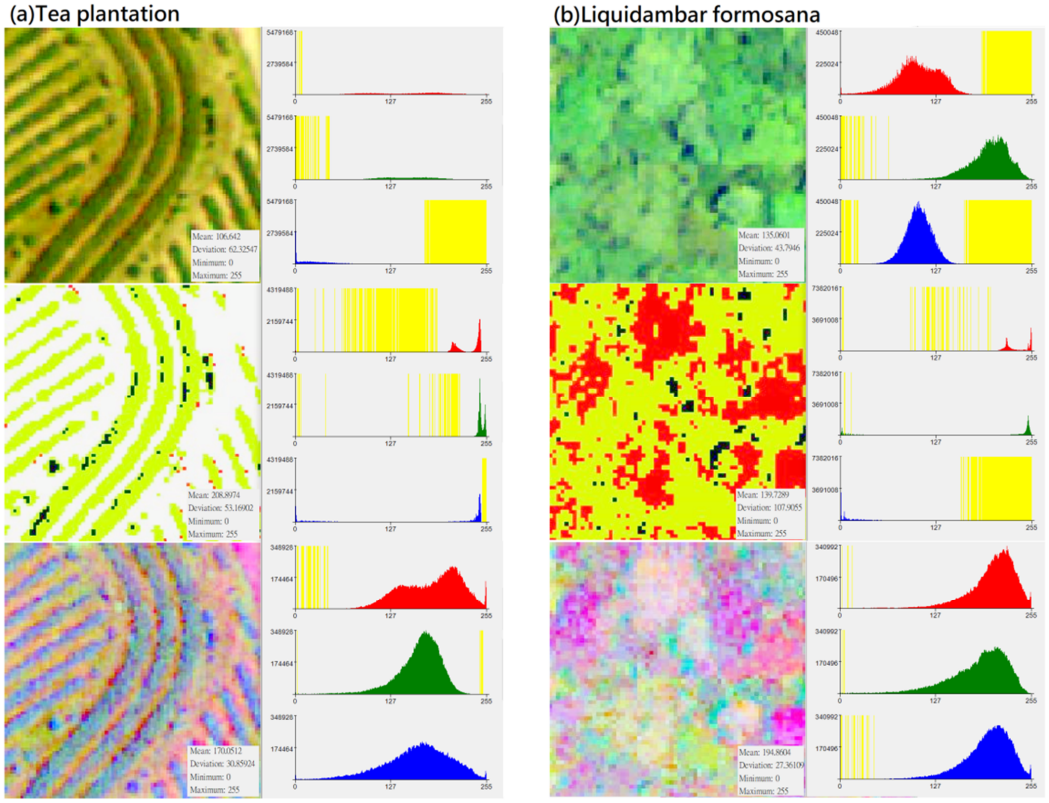


Figure 3. *Cont.*

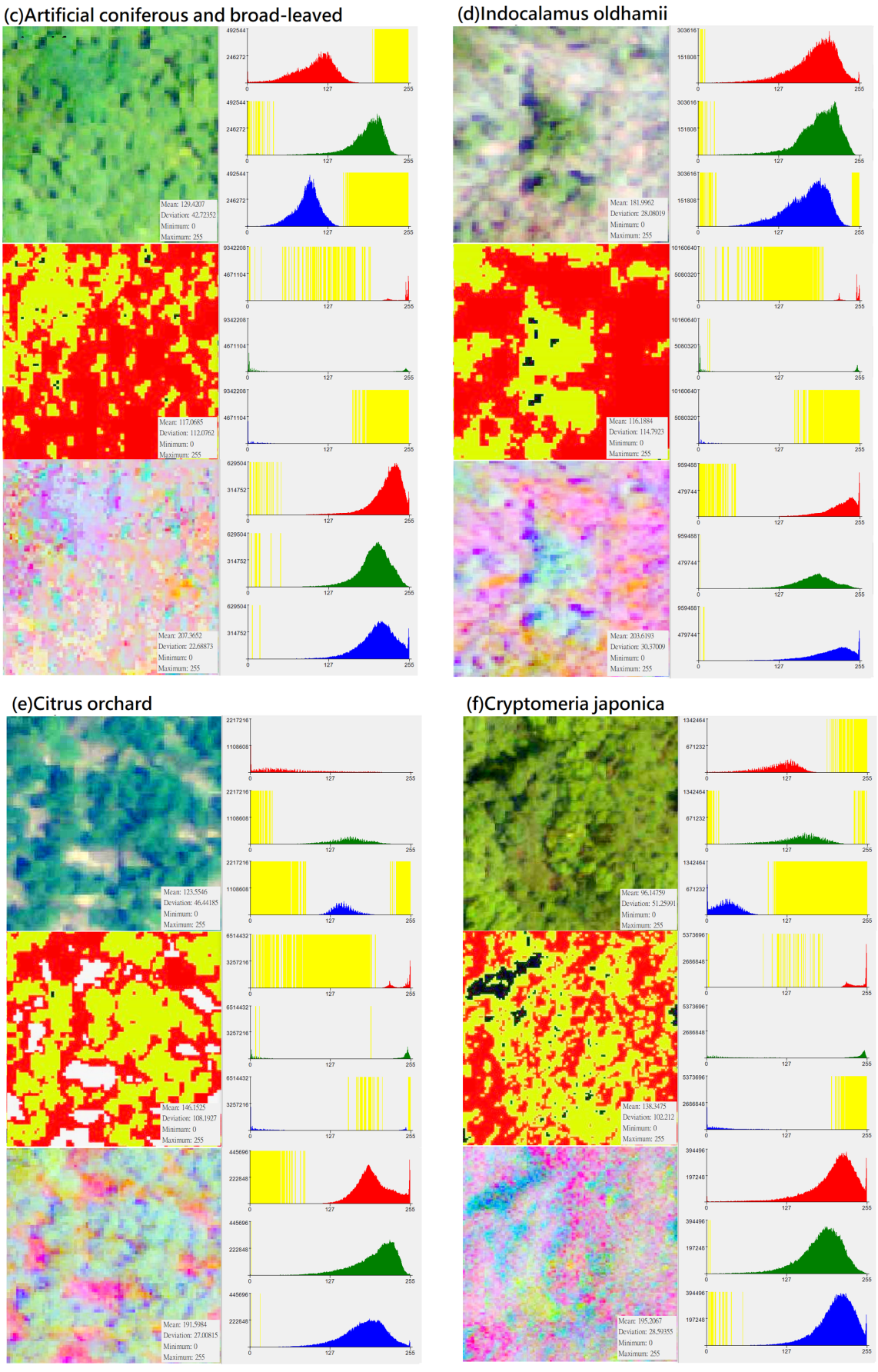


Figure 3. *Cont.*

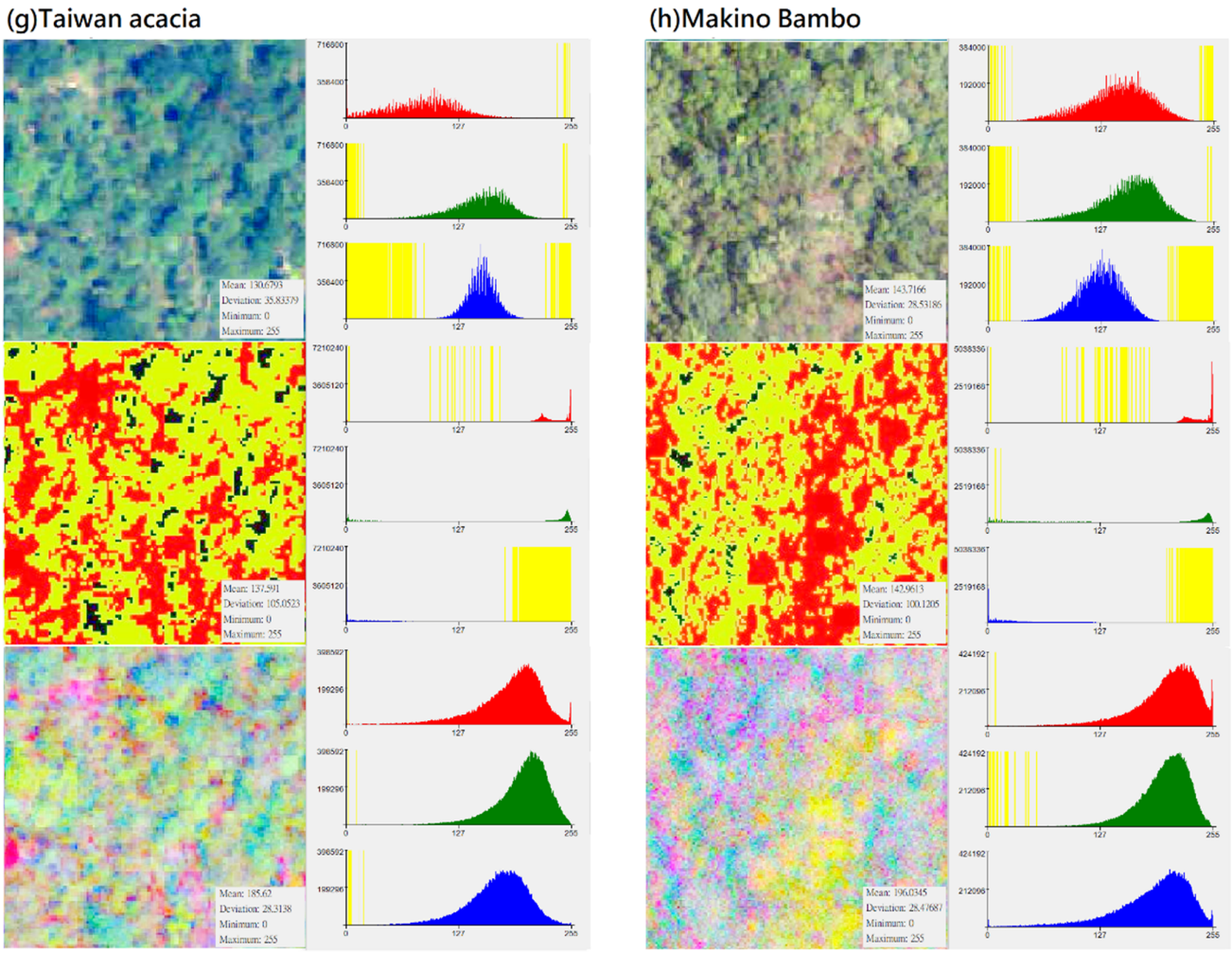


Figure 3. Distribution of DN value (X-axis) and pixel number (Y-axis) of different vegetation types in "original photographs", "CS-based aerial photographs", "PCA-based aerial photographs" (yellow represents the zero value area, subfigures (**a**–**h**) represent several types of forest and vegetation types that this research mainly wants to identify).

This study used principal-axis transformation to perform PCA transformation on multi-band datasets. Since the RGB bands of original aerial photographs usually have correlations (for example, a positive correlation or a negative correlation between DN values in a certain range), the applicability of the data is also reduced. PCA is a linear transformation that reorganizes the variance in a multi-band image into a set of new image bands and removes redundant spectral information from the multi-band dataset. These PCA bands are uncorrelated linear combinations of the input bands. In the transformation process, the PCA transformation finds a new set of orthogonal axes (x: red belt, y: green belt, z: blue belt), the origin is located at the data mean, and they are rotated to maximize the data variance.

This study performed the following steps to perform PCA:

1. Entered the image covariance or correlation matrix, and calculated the eigenvector of the correlation matrix. The eigenvector matrix showed the statistical correlation between the PC (dependent variable or row) and the input image band (independent variable or column). The feature vector itself represented the contribution ratio of each input image band to each PCA band. The weight of each input band was calculated by squaring the eigenvector elements of the input band. Therefore, the total contribution of all input bands to any given PCA band was the sum of the squares of the eigenvector elements of the PCA band.

2. Subtracted the band average value from the input image data. This correction produced an origin shift in the output PCA image so that the average spectrum of each band was 0.
3. Used the following equation to project the mean-corrected image data onto the transpose of the eigenvector matrix:

$$y = G \# (\text{average value of } x) \quad (1)$$

where:

y = transformed (or rotated) data
G = transformation matrix
x = input data
# Indicates matrix multiplication

In this study, the number of input bands is the same as the output PCA band. The first PCA band contains the largest percentage of data variance, and the second PC band contains the second largest data variance. The final PCA bands look noisy because it contains very little variance, most of which comes from the noise in the raw spectral data. Since the data is uncorrelated, the PCA band produces more color composite images than spectral color composite images (Table 11). It can be seen from Figure 3 that after the PCA conversion, the number of grids corresponding to the DN value of each band in the new layer is transformed into an approximately normal distribution, and the difference in the number of image pixels is much reduced (e.g., extreme values are eliminated). PCA conversion can effectively eliminate the problem of unequal weights among the three RGB bands and the problem of extreme values in the number of pixels on the *Y*-axis. Since the data is normalized and extreme values do not exist, the weight difference of different DN values applied to classification in each band is reduced, which increases the efficiency of data application to a certain extent. This conversion result can be regarded as an increase in the number of features for the CNN model as a basis for learning and an increase in the probability of being sampled. Therefore, PCA is generally expected to improve the efficiency of subsequent applications. During the analysis process, this study performed CS-based and PCA-based image conversion for the graphics of the entire study area, and then uniformly extracted the training group and testing group images of various different vegetation types from the entire area. Therefore, every image for training and testing had undergone the same and consistent conversion process. In other words, there was no issue of an essential difference between the training sample and the test data.

2.4. Classification of Forestland Using Deep Learning and Effectiveness Evaluation

Considering the superior performance of deep learning methods over existing methods in computer vision problems, such as road traffic accident avoidance [36], semantic segmentation [37], and classification and detection [38], we used a deep learning network and an end-to-end open source platform called TensorFlow for forestland classification. TensorFlow, which was created by the Google Brain team, is a machine learning framework used for designing, building, and training deep learning models. It is also an open source library that allows the public to perform numerical computations.

After conducting preprocessing on the original aerial photographs, a deep learning calculation module was used for forestland classification. The preliminary feature map of a CNN may include simple and obvious shape features, but as the network gradually deepens, the convolution process can reveal more complex patterns and higher-level concepts, such as spatial distribution characteristics, semantic associations, and feature combinations, thereby increasing the feasibility of classification.

However, CNN-based training is performed using a data array called tensor (including multiple dimensions). In the training process, a CNN achieves detection by transforming the input data through convolutions and other hidden layers while propagating through the network to determine whether the detection results between the layers are activated as neurons or simulated (such as ReLU). If the detection result is an activation, the activation

output intensity depends on its weight and bias and can be optimized using a gradient descent algorithm until the model performance stops improving (model convergence) or declines (model overfitting). The simulation results in this part depend on various factors, such as CNN model input data, model complexity, and model structure. A CNN can set up completely different architectures according to task requirements, such as a model suitable for object detection, a model suitable for spatial image classification, and a model suitable for semantic segmentation. Whether these models are applicable to the needs of a specific field needs to be tested.

This study employed the SegNet, ResNet50, and VGG19 modules and compared the forestland classification accuracies of each method. The differences between the modules are detailed in the following sections.

2.4.1. VGG19

Training CNNs for image classification requires comparably simple annotations in the form of class correspondences or continuous values, respectively, for each image. Therefore, if more subtle features or semantic meanings can be extracted, the efficiency and accuracy of classification will be effectively improved.

In 2014, the VGG Laboratory at Oxford University was the runner-up in the ImageNet classification task at ILSVRC14. The VGG Lab proposed a series of network models, such as VGG11, VGG13, VGG16, and VGG19, and increased the network depth to a maximum of 19 layers [25].

The structure of VGG19 is very simple: the entire network uses the same convolution kernel size (3×3) and maximum pooling size (2×2). However, conceptually, a combination of several small-filter (3×3) convolutional layers is more effective than a large-filter (5×5 or 7×7) convolutional layer. Single image classification was performed at the end of a layer with softmax activation. This softmax activation mapped the learned features (original, CS-based, and PCA-based photographs of seven or 14 forest and vegetation types) to the final class probabilities. The maximum class probability of a pixel represented the final class of the respective pixel. We then applied an argmax function to obtain the class label for the class with the highest probability. This model also verified that the classification performance could be improved by continuously deepening the network structure, but this can result in the waste of more computing resources.

2.4.2. ResNet50

For a traditional CNN, the deeper the depth of the convolution layer, the better the recognition effect; however, problems such as decreased performance and overfitting can then occur. Overfitting is caused by the decrease in both the training and verification accuracies, which reveals that the difficulty of training a deep learning network increases with the network's depth.

Many deep learning studies have made small improvements by adjusting the mathematics, optimization, and training processes without considering the underlying tasks of the model. ResNet, which was proposed by He et al. [27], has fundamentally changed the understanding of neural networks and how they learn.

The VGG19 network architecture builds on and modifies the ResNet network, and a residual unit involving a short-circuit mechanism was added to solve the vanishing gradient problem and reduced accuracy problem (training set) in the deep network, thereby enabling the deepening of the network. Thus, ResNet not only ensures high accuracy but also controls the speed. VGG19's redundant layers are meant to complete the identity mapping and ensure that the input to and output from the identity layer are exactly the same. In contrast, its specific layers are the identity layers that are determined by network training. Several layers of the original network are modified into a residual block, and more details about this architecture can be found from He et al. [27]. A residual block has the same main architecture as the original network architecture but also has a shortcut link. This shortcut link skips a layer of linear transformation and non-linear output and directly

becomes the input of the activation function in the upper two layers. Since this shortcut link provides the output of the source layer with a link that can transform the identity or create copies, the non-adjacent target layer reduces the conversion of one layer, so it is also called a skip connection. The mode setting adopts the source layer output that is directly copied through the skip connection. The advantage of this is that if the target layer has unnecessary parameters (for example, when a regularization item is added), the weight and offset parameters of the target layer will be reduced to zero. However, since the output of the target layer is a nonlinear conversion of the output of the source layer, the problem of gradient disappearance will not occur. We added one fully connected layer after average pooling at the end of ResNet50 according to different classification situations (seven or 14 classifications). This allowed us to make them more appropriate for our data and finish the final classifier. We used the ideas from the ResNet-based tree species classification but utilized a different structure by using UAV images by Natesan et al. [39].

2.4.3. SegNet

SegNet uses the concept of semantic segmentation to segment the desired image area and recognize its content. The convolutional layer is used to extract the features in the image, and then the maximum pooling and interpolation information is repeatedly used for the samples lost in the other sampling process through the encoder and the corresponding decoder to obtain the samples. Therefore, compared to other semantic segmentation methods, SegNet provides a greater reduction in the number of training parameters and an increase in the number of image contours. In the network structure of SegNet, the encoder alternately uses convolution and pooling, and the decoder alternately uses deconvolution and upsampling. We used softmax for pixel-based classification to obtain the final class label for the class with the highest probability. In the encoder–decoder process, pooling indicators (e.g., location information during pooling) are used to transfer the decoder to improve the image segmentation rate.

The maximum pooling in the above process can realize the invariance of translation when a small spatial displacement is performed on the input image. If the memory in the inference process is not constrained, all encoder feature maps (after downsampling) can be stored, and SegNet proposes an effective method to store this information. It only stores the maximum pooling index (the location of the largest feature value in each pooling window), which is used for each encoder feature map.

2.5. Evaluation of Different CNN Models

We used mature models and the VGG 19, ResNet50, and SegNet modules with different types of images. The effectiveness of the models was verified. A confusion matrix was first generated from the predicted and actual categories. The matrix contains four elements (Figure 4): true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN). Determining the effectiveness of the models using these values would be difficult; therefore, indicators such as recall, precision, and F1 score were used to evaluate the quality of the models. The formulas are shown in (2), (3), and (4).

$$\text{Recall} = \text{TP}/(\text{TP}+\text{FN}) \quad (2)$$

$$\text{Precision} = \text{TP}/(\text{TP}+\text{FP}) \quad (3)$$

$$\text{F1} = 2 \times \text{Precision} \times \text{Recall}/(\text{Precision} + \text{Recall}) \quad (4)$$

The recall rate is the proportion of all positives that can be predicted. The precision rate is the percentage of complete TP matches out of all TP matches. For example, what is the significance of a model with high precision but low recall and a model with high recall but low precision? The former can be regarded as a more cautious model. Although there is often no way to capture positives, as long as they are captured, they are almost always correct (precision is high). While the latter is a loose model that sometimes catches false TPs, almost every positive can be caught (recall high). Both of the above models

perform poorly in extreme situations. The former may rarely predict a positive sample, while the latter may have too many prediction errors. Therefore, we used F1-score, which represents the harmonic average of the two and is considered a more comprehensive model evaluation index.

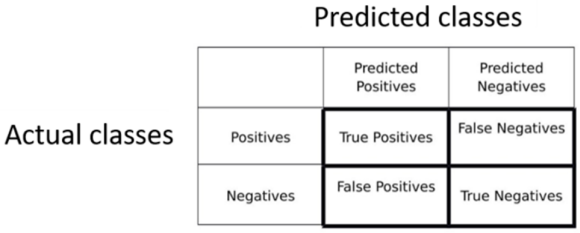


Figure 4. Confusion matrix.

2.6. Computational Environment

The hardware environment of the prototype web-based system developed in this paper was as follows; programming language: Python, operating system: Windows 10, graphics card: RTX 2080Ti GDDR6 11 GB, processor: Intel (R) Core (TM) i7-9700KF CPU @ 3.60 GHz, and memory: DDR4 3000 MHz 32 GB. Regarding software, we used Flask, a lightweight web application framework written in Python. Its core is mainly composed of the Werkzeug WSGI toolbox and Jinja2 template engine. Matplotlib is an important module for data visualization in Python and was used in the system. It can draw various charts, it has complete drawing functions, and it has syntax similar to that of MATLAB. NumPy is an extended library for Python, and its bottom layer is implemented in C and Fortran. It supports high-order dimensional matrix operations and provides numerous multidimensional matrix mathematical function libraries. When Python conducts large matrix operations, the processing performance of Python List is not ideal. NumPy has superior parallel processing capabilities and can thus operate various functions and operators more directly and efficiently, which greatly improves execution efficiency (Available online: https://scipy.github.io/old-wiki/pages/ParallelProgramming; Accessed date: 1 May 2021). The Python libraries used in this experiment are listed in Table 2. To allow everyone to reproduce our work, we made the code used in this paper available online: https://github.com/fclin666/Tree-Species-Identification(Link establishment date: 1 July 2021). Additionally, the different model parameter sizes of the three models used in this paper are shown in Table 3.

Table 2. The versions of Python libraries used in this research.

Python Library	Version
TensorFlow-gpu	2.0.0-rc0
Keras	2.3.1
Numpy	1.18.1
Matplotlib	3.1.3
sklearn	0.22.1
pandas	1.0.3

Table 3. Model parameter sizes used in this research.

	VGG19 (Model: "Sequential")	ResNet50	SegNet
Total params	20.024.384	23.561.152	31.821.646
Trainable params	20.024.384	23.508.032	31.804.750
Non-trainable params	0	53.120	16.896

3. Results

3.1. The Results of Applying VGG19, ResNet50, and SegNet Models on Original Aerial Photographs

For indicator verification, VGG19, ResNet50, and SegNet models were used after simulations to determine the correct rate through interpretation (the cumulative number of correct judgments divided by 500). Moreover, a loss function indicator was employed to assess the models' stability. Table 4 presents the results obtained using VGG19, ResNet50, and SegNet for the 14 forest and vegetation types (original images). Table 5 lists the results obtained using VGG19, ResNet50, and SegNet for the seven forest and vegetation types (Types 1, 2, 4, 7, 8, 10, and 11; original images). Finally, the computational time and accuracy of the analyses of the original aerial photography using VGG19, ResNet50, and SegNet are summarized in Table 6.

Table 4. Precision, recall, and F1 score of original aerial photography when using VGG19, ResNet50, and SegNet in 14 types of images (ID number: (1) Artificial coniferous and broad-leaved mixed forest, (2) Natural coniferous and broad-leaved mixed forest, (3) Natural grasslands, (4) Natural broad-leaved mixed forest, (5) Dryland farming, (6) Artificial pine forest, (7) Citrus orchard, (8) *Cryptomeria japonica*, (9) Taiwan acacia, (10) Makino Bamboo, (11) Tea plantation, (12) *Liquidambar formosana*, (13) *Indocalamus oldhamii*, and (14) *Cunninghamia lanceolata*).

Forest and Vegetation Types (Original Photographs)									
	VGG19 (Accuracy: 93%)			ResNet50 (Accuracy: 66.00%)			SegNet (Accuracy: 82.00%)		
ID	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1
1	1.00	0.98	0.99	0.45	0.86	0.59	0.67	1.00	0.80
2	0.90	0.90	0.90	1.00	0.65	0.79	1.00	0.75	0.86
3	1.00	1.00	1.00	0.00	0.00	0.00	1.00	1.00	1.00
4	1.00	1.00	1.00	1.00	0.84	0.91	0.89	1.00	0.94
5	0.69	1.00	0.81	0.27	0.71	0.40	1.00	0.89	0.94
6	1.00	1.00	1.00	0.00	0.00	0.00	0.59	0.96	0.73
7	0.71	0.89	0.79	0.40	1.00	0.57	0.00	0.00	0.00
8	0.84	0.95	0.89	0.94	1.00	0.97	1.00	1.00	1.00
9	0.95	0.69	0.80	0.88	0.53	0.66	1.00	1.00	1.00
10	1.00	1.00	1.00	0.64	1.00	0.78	0.63	1.00	0.77
11	1.00	1.00	1.00	0.88	1.00	0.94	1.00	0.62	0.76
12	1.00	1.00	1.00	0.82	1.00	0.90	1.00	0.31	0.48
13	1.00	1.00	1.00	0.44	0.50	0.47	0.95	0.43	0.59
14	0.95	0.81	0.88	1.00	0.84	0.91	0.85	0.66	0.74

Table 5. Precision, recall, and F1 score of original aerial photography when using VGG19, ResNet50, and SegNet.

Seven Forest and Vegetation Types (Type 1, Type 2, Type 4, Type 7, Type 8, Type 10, and Type 11) (Original Photographs)									
	VGG19 (Accuracy: 98%)			ResNet50 (Accuracy: 81%)			SegNet (Accuracy: 95.65%)		
ID	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1
1	0.98	1.00	0.99	0.88	0.26	0.40	0.85	1.00	0.92
2	1.00	0.89	0.94	1.00	1.00	1.00	1.00	1.00	1.00
4	1.00	1.00	1.00	0.57	1.00	0.73	1.00	1.00	1.00
7	0.90	1.00	0.95	0.66	1.00	0.79	1.00	0.40	0.40
8	1.00	0.93	0.97	1.00	0.96	0.98	1.00	1.00	1.00
10	0.80	1.00	0.89	1.00	1.00	1.00	1.00	1.00	1.00
11	1.00	1.00	1.00	1.00	1.00	1.00	1.00	0.99	0.99

According to the results of Tables 4 and 5, if the original aerial photographs were used as data input, the classification accuracy of the three models was VGG19 > SegNet > ResNet50, and the results of the seven forest and vegetation types were significantly better than those of the 14 forest and vegetation types. This shows that although both ResNet50 and SegNet are improved models that came out after VGG19, they could not

achieve classification advantages in identifying forests that were repeatedly distributed over a large area and had very similar aerial photograph characteristics. Additionally, the data in Table 4 show that the recall rates of natural grasslands, Taiwan acacia, artificial pine forest, and *Indocalamus oldhamii* through ResNet50 were extremely low (0%, 0%, 53%, and 50%). The proportion of positives that were successfully predicted was low, and there may even be cases where all of them were classified into other categories. Since ResNet50 mainly uses residual learning instead of the original layer full-feature learning method to increase the depth of feature extraction, it is obviously not suitable for forest types with finer leaf textures and continuous distribution. Identity mapping may thus increase the differences with other plant types due to the coincidence of low-level features. In addition, regarding classification accuracy, ResNet50 also had the problem of commission error. If various features in the original image are highly similar and overlap with each other, it is not suitable to use a model that increases residual learning.

However, the results of SegNet's classification of original aerial photographs were not as good as expected, with results of "high precision and low recall" and "high recall and low precision," respectively. Since SegNet mainly obtains and stores boundary information in the feature map of the encoder before downsampling, it reduces the possible size of the feature map and loss of spatial information. However, there were no obvious boundary details in the forest and vegetation categories to be classified in this research, so the model's overall performance was not as good as that of VGG19. Additionally, the classification effect of fewer categories was better than that of more categories.

As shown in Table 6, when 14 surface coverage classifications were determined from the original aerial photographs, the training time of SegNet was much longer than that of VGG19 or ResNet50, and it was the same as the time required to perform classification calculations. This is because VGG19 and ResNet50 are directly improved models of CNN and are only adjusted for the original structure. SegNet adds location information storage for boundary features, so its system resources and computing costs are greater.

In terms of model operation, the VGG19 and SegNet models were relatively more stable than ResNet50. This study found that increasing the depth of the network did not effectively improve the classification results. In this simulation, ResNet50 was the only model that achieved an accuracy of less than 90% (66% for 14 types and 81% for seven types), which indicates that the customized improvement of this model is not suitable for large-scale forestland classification. In addition, due to the unstable fluctuation characteristics of the loss function curve, the model could not extract the representative forest or vegetation characteristics from the aerial photographs during the training process, which also led to its inability to obtain stable results.

Table 6. Precision, recall, and F1 score of original aerial photography when using VGG19, ResNet50, and SegNet.

Fourteen Forest and Vegetation Types			
Algorithm	**Training Time (30 Epochs)**	**Testing Time (500 Photographs)**	**Accuracy**
VGG19	275.84 s	5.48 s	0.93
ResNet50	778 s	44 s	0.66
SegNet	1540 s	57 s	0.82
Seven Forest and Vegetation Types (Type 1, Type 2, Type 4, Type 7, Type 8, Type 10, and Type 11)			
Algorithm	**Training Time (30 Epochs)**	**Testing Time (231 Photographs)**	**Accuracy**
VGG19	121.56 s	2.73 s	0.98
ResNet50	378 s	40 s	0.81
SegNet	681 s	26 s	0.95

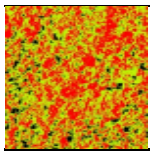
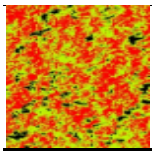
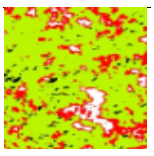
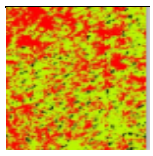
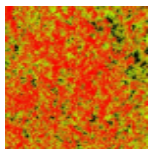
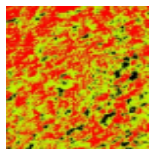
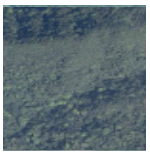
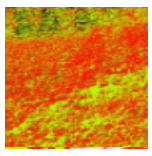
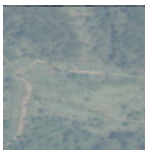
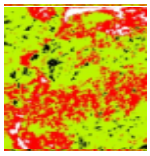
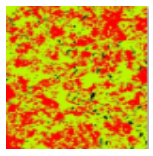
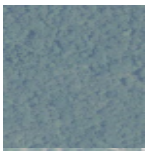
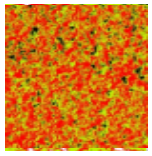
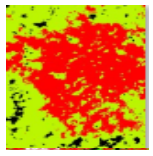
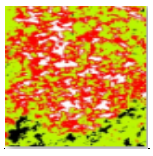
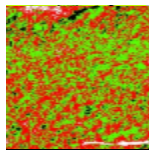
3.2. Effectiveness of the CNN Models When Using CS-Based Photographs

According to the results of the previous paragraph, when using the original aerial photographs with the CNN models, the models tended to filter useful information. This

study attempted to use content-simplified new inputs with four categories: light surface of the tree canopy, shadow surface of the tree canopy, gaps between trees, and bare ground. Table 7 shows a preliminary classification training sample for converting the original aerial photographs into CS-based photographs (a total of 14 forest and vegetation types).

Table 8 shows the results obtained using VGG19, ResNet50, and Table 9 presents the results obtained using VGG19, ResNet50, and SegNet for the 14 forest and vegetation types (CS-based photographs), and SegNet for the seven forest and vegetation types (CS-based photographs). The computational time and accuracy for CS-based photographs when using VGG19, ResNet50, and SegNet are summarized in Table 10.

Table 7. Fourteen types (original aerial photograph and CS-based photographs) (ID number: (1) Artificial coniferous and broad-leaved mixed forest, (2) Natural coniferous and broad-leaved mixed forest, (3) Natural grasslands, (4) Natural broad-leaved mixed forest, (5) Dryland farming, (6) Artificial pine forest, (7) Citrus orchard, (8) *Cryptomeria japonica*, (9) Taiwan acacia, (10) Makino Bamboo, (11) Tea plantation, (12) *Liquidambar formosana*, (13) *Indocalamus oldhamii*, and (14) *Cunninghamia lanceolata*).

Forest and Vegetation Types	Original Aerial Photograph	CS-Based Aerial Photograph	Forest and Vegetation Types	Original Aerial Photograph	CS-Based Aerial Photograph
(1)			(8)		
(2)			(9)		
(3)			(10)		
(4)			(11)		
(5)			(12)		
(6)			(13)		
(7)			(14)		

The operating concept of this part of the study was a bit similar to "Dropout" in the regularization technology [40], but our research did not artificially introduce randomness during the model operation and instead reduced the statistical noise during the data

input and training process. The table contents show that if CS-based photographs of 14 forest and vegetation types were used for input, the classification accuracy of the three models was still in the order of VGG19 > SegNet > ResNet50. It is worth noting that if the classification category was reduced from 14 to 7, the classification accuracy ranking was SegNet > VGG19 > ResNet50. This result shows that even if the amount of input information is reduced, ResNet50 can still obtain a lower recall rate of specific types due to the continuous distribution and highly repetitive vegetation characteristics, and it also affects the proportion of positives.

Take artificial coniferous and broad-leaved mixed forest (Type 1), natural broad-leaved mixed forest (Type 4), *Cryptomeria Japonica* (Type 8), and Makino Bamboo (Type 10) as examples. We found that the performance in the recall value was poor because the feature distribution of every type in the simplified CS-based photographs was very similar. Moreover, there was no obvious edge feature or boundary shape in CS-based photographs for ResNet50 to perform effective residual learning, so that also affected the performance of the recall value.

However, in terms of data comparison between SegNet and VGG19, this study found that the performance of the two models became more similar after reducing the information complexity of the data input. However, whether the classification category was 7 or 14, the two models still displayed some commission and omission errors in category determination (the recall value was low). We examined the categories with poor classification performance and found that the spatial features between different categories were similar, and the arrangement of these simplified features was mostly random and irregular (e.g., Type 1, Type 3, Type 10). Therefore, even if the SegNet model had stored the boundary information of the feature map, the classification of forest plants with irregular canopy arrangements may still have caused misjudgment. According to Table 10, SegNet spent much more time training and performing classification calculations than VGG19 and ResNet50. In general, only VGG19 could achieve an accuracy rate as high as 93%, which shows that when overly simplified feature layers are used, the strength of CNNs—their extraction of detailed features and additional characters—cannot be applied, and the results are affected by the subjective extraction of feature layers. However, the loss function curve revealed that the use of feature layers greatly improved the stability of repeated model prediction; therefore, some benefits were retained for CNN calculations.

Table 8. Precision, recall, and F1 score of CS-based photographs when using VGG19, ResNet50, and SegNet.

Seven Forest and Vegetation Types (CS-Based Photographs)									
	VGG19 (Accuracy: 95%)			**Resnet50 (Accuracy: 60%)**			**Segnet (Accuracy: 95.69%)**		
	Precision	**Recall**	**F1**	**Precision**	**Recall**	**F1**	**Precision**	**Recall**	**F1**
1	1.00	0.98	0.99	0.32	0.39	0.35	0.88	1.00	0.94
2	1.00	1.00	1.00	1.00	0.80	0.89	1.00	0.82	0.90
4	1.00	1.00	1.00	0.43	0.94	0.59	1.00	1.00	1.00
7	1.00	1.00	1.00	1.00	1.00	1.00	0.87	0.87	0.87
8	0.81	0.98	0.89	0.94	0.30	0.45	1.00	0.91	0.96
10	0.88	0.41	0.56	0.00	0.00	0.00	1.00	0.87	0.93
11	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00

Table 9. Precision, recall, and F1 score of original aerial photography when using VGG19, ResNet50, and SegNet for 14 types of images (CS-based photographs).

CS-Based Photographs (14 Forest and Vegetation Types)									
	VGG19 (Accuracy: 93%)			Resnet50 (Accuracy: 32%)			Segnet (Accuracy: 87.20%)		
	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1
1	0.89	0.86	0.88	0.00	0.00	0.00	0.82	1.00	0.90
2	1.00	1.00	1.00	0.43	1.00	0.60	1.00	0.88	0.94
3	0.92	0.94	0.93	0.19	0.44	0.27	1.00	0.12	0.22
4	0.87	1.00	0.93	0.14	0.91	0.25	0.88	1.00	0.93
5	0.86	1.00	0.92	0.83	0.54	0.66	0.88	0.82	0.85
6	0.88	0.92	0.90	0.00	0.00	0.00	1.00	0.93	0.96
7	1.00	1.00	1.00	1.00	0.53	0.70	0.28	1.00	0.44
8	0.86	0.91	0.89	0.00	0.00	0.00	0.96	0.93	0.95
9	1.00	0.91	0.95	1.00	0.08	0.14	0.89	0.91	0.90
10	0.75	0.71	0.73	0.00	0.00	0.00	1.00	0.84	0.91
11	1.00	1.00	1.00	0.95	1.00	0.98	1.00	0.92	0.96
12	1.00	0.96	0.98	0.00	0.00	0.00	1.00	1.00	1.00
13	1.00	0.93	0.97	0.54	0.82	0.65	1.00	0.90	0.95
14	1.00	0.87	0.93	0.67	0.05	0.09	0.98	1.00	0.99

Table 10. Computational time and accuracy for feature-based images when using VGG19, ResNet50, and SegNet.

Fourteen Forest and Vegetation Types			
Algorithm	**Training Time (30 Epochs)**	**Testing Time (500 Images)**	**Accuracy**
VGG19	285.52 s	5.71 s	0.93
ResNet50	777 s	44 s	0.32
SegNet	1488 s	57 s	0.87
Seven Forest and Vegetation Types			
Algorithm	**Training Time (30 Epochs)**	**Testing Time (231 Images)**	**Accuracy**
VGG19	127.61 s	2.89 s	0.95
ResNet50	371 s	40 s	0.60
SegNet	711 s	19 s	0.96

3.3. Effectiveness of the Models When Using Forestland PCA-Based Photographs

PCA-based photographs were the last data input method in this study. Table 11 shows the preliminary classification training samples for converting the original aerial photographs into PCA-based images (a total of 14 forest and vegetation types). The data in Tables 12–14 show that when the seven categories were classified, the order of the classification accuracy of the three models was SegNet > ResNet50 > VGG19. In the case of 14 types of classification, the accuracy ranking was SegNet > VGG19 > ResNet50, and the value reached more than 90%. This indicated that if the original aerial photographs were first converted to PCA-based photographs, the result, particularly the recall rate, could greatly improve no matter what kind of model or category was applied. This also meant that the proportion of predicted positives was very high, and specific types of forest and vegetation that were searched were rarely classified into other types. Compared to the previous two types of data inputs, the data input based on PCA-based images could greatly improve the operational effectiveness of ResNet50. The reason may be that after the data dimension was reduced, the independence of the data features in the original image increased. The correlations among multiple features within photographs were also removed by PCA, which enhanced the feature extraction ability when performing residual learning. A similar situation also affected the SegNet model. Since PCA-based photographs reduce the relationship between features and features within photographs, their boundary definitions for distinguishing features are clearer and more obvious, which improves

the segmentation ability between categories. This study thus found that unless aerial photographs have been properly pre-processed as input files, even the specially improved CNN model may not perform better than the original models in terms of classification accuracy. However, in terms of computing time, the improved ResNet50 and SegNet still consumed much more time than VGG19.

Table 11. Fourteen forest and vegetation types (original aerial photograph and PCA-based photograph) (ID number: (1) Artificial coniferous and broad-leaved mixed forest, (2) Natural coniferous and broad-leaved mixed forest, (3) Natural grasslands, (4) Natural broad-leaved mixed forest, (5) Dryland farming, (6) Artificial pine forest, (7) Citrus orchard, (8) *Cryptomeria japonica*, (9) Taiwan acacia, (10) Makino Bamboo, (11) Tea plantation, (12) *Liquidambar formosana*, (13) *Indocalamus oldhamii*, and (14) *Cunninghamia lanceolata*).

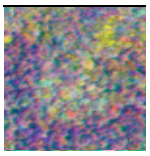
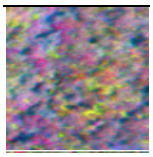
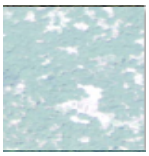
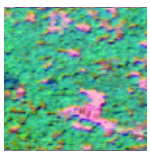
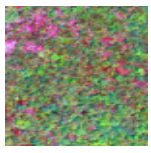
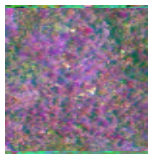
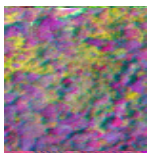
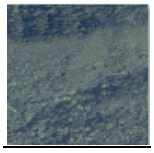
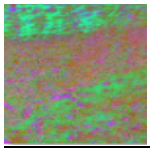
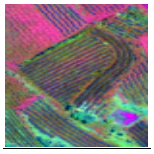
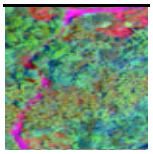
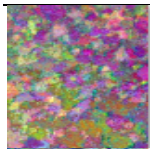
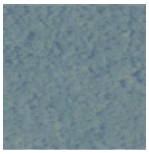
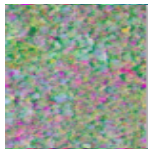
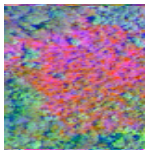
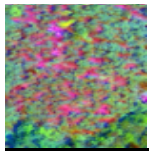
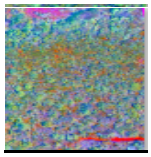
Forest and Vegetation Types	Original Aerial Photograph	PCA-Based Aerial Photograph	Forest and Vegetation Types	Original Aerial Photograph	PCA-Based Aerial Photograph
(1)			(8)		
(2)			(9)		
(3)			(10)		
(4)			(11)		
(5)			(12)		
(6)			(13)		
(7)			(14)		

Table 12. Precision, recall, and F1 score of original aerial photography when using VGG19, ResNet50, and SegNet in 14 types of images (PCA-based photographs).

Fourteen PCA-Based Photographs									
	VGG19 (Accuracy: 96%)			ResNet50 (Accuracy: 93%)			SegNet (Accuracy: 98.20%)		
	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1
1	0.93	1.00	0.96	1.00	0.75	0.85	0.97	1.00	0.98
2	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
3	1.00	0.86	0.93	0.98	1.00	0.99	1.00	0.97	0.98
4	1.00	1.00	1.00	0.95	1.00	0.97	0.93	1.00	0.96
5	1.00	1.00	1.00	0.88	1.00	0.94	0.95	0.91	0.93
6	0.75	1.00	0.86	1.00	1.00	1.00	1.00	0.98	0.99
7	0.93	1.00	0.96	1.00	1.00	1.00	0.92	1.00	0.96
8	0.98	0.85	0.91	0.98	0.91	0.95	0.98	1.00	0.99
9	1.00	1.00	1.00	0.96	0.89	0.92	1.00	1.00	1.00
10	0.91	0.83	0.87	1.00	0.58	0.74	1.00	1.00	1.00
11	1.00	1.00	1.00	1.00	0.97	0.98	1.00	1.00	1.00
12	1.00	1.00	1.00	0.74	1.00	0.85	1.00	1.00	1.00
13	1.00	0.95	0.97	1.00	0.95	0.98	1.00	0.91	0.96
14	0.93	0.98	0.96	0.79	1.00	0.88	0.97	0.98	0.98

Table 13. Precision, recall, and F1 score of PCA-based images when using VGG19, ResNet50, and SegNet for seven forest and vegetation types (PCA-based photographs).

	VGG19 (Accuracy: 92%)			ResNet50 (Accuracy: 96%)			SegNet (Accuracy: 100%)		
	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1
1	1.00	1.00	1.00	1.00	0.93	0.96	1.00	1.00	1.00
2	1.00	1.00	1.00	0.92	1.00	0.96	1.00	1.00	1.00
4	1.00	1.00	1.00	0.98	1.00	0.99	1.00	1.00	1.00
7	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
8	1.00	0.60	0.75	1.00	0.89	0.94	1.00	1.00	1.00
10	0.39	1.00	0.56	0.71	1.00	0.83	1.00	1.00	1.00
11	1.00	1.00	1.00	0.94	1.00	0.97	1.00	1.00	1.00

Table 14. Computational time and accuracy for PCA-based images when using VGG19, ResNet50, and SegNet.

Fourteen Forest and Vegetation Types			
Algorithm	**Training Time (30 Epochs)**	**Testing Time (500 Images)**	**Accuracy**
VGG19	280.91 s	5.58 s	0.96
ResNet50	778 s	44 s	0.93
SegNet	1497 s	57 s	0.98
Seven Forest and Vegetation Types			
Algorithm	**Training Time (30 Epochs)**	**Testing Time (500 Images)**	**Accuracy**
VGG19	130.17 s	2.64 s	0.92
ResNet50	367 s	40 s	0.96
SegNet	670 s	19 s	1.00

3.4. Accuracy Iteration and Loss Iteration of Different Image and Model Combinations

To calculate the efficiency of different input image and model combinations, this study used accuracy and loss iterations with the default value set to 30 epochs. Figure 5 shows the results. Figure 5a shows that when using the VGG19 model, the accuracy of the green part (PCA-based image) was higher than that of either the blue (original image) or red parts (CS-based image). Furthermore, the verification group was better than the training group. It showed that if the input data were processed by PCA first, this could effectively improve the model's accuracy. For Figure 5b, we can see that each curve showed a convergence

trend, and the decline rate of the green part (PCA-based image) was significantly better than that of either the blue (original image) or red parts (CS-based image), but the difference was not obvious. On the whole, no matter what input image was used, the learning rate for the first half (0–15 epochs) of the VGG19 model was quite high, and the curve quickly converged, which is undoubtedly a safe and stable choice for users.



Figure 5. Accuracy and loss iterations of different image and model combinations when using seven forest and vegetation types (subfigures (**a**,**c**,**e**) represent accuracy iterations of VGG19, ResNet50 and SegNet; subfigures (**b**,**d**,**f**) represent loss iterations of VGG19, ResNet50 and SegNet).

Regarding the accuracy and loss iterations of ResNet50 (Figure 5c,d), no matter the original image, when using CS- or PCA-based images, the curve was unstable, and the loss curve showed a slow linear decline and low learning rate. Taking the CS-based image as an example, its accuracy varied in the range 0.5–0.9 in 10–30 epochs. In the loss rate part, no matter what kind of input image was used, continuous and stable convergence was not

shown, but when the training reached the 30th epoch, it showed that the performance of the PCA-based image was superior to that of the CS-based and original images. Furthermore, the trend of curve convergence was obvious. This meant that even if the simulation results were not good due to the influence of the model settings, PCA pre-processing for the image still effectively stabilized the model's classification ability.

In Figure 5c, the represented CS-based image had a special impact on SegNet's accuracy iteration due to the removal of too much available information. Whether the model was in the training or verification process, the accuracy rate only reached close to 90% in the end. However, the decline slope of the loss function curve was also obviously gentler, which meant that the model's learning ability was obviously limited. The accuracy curve and loss function curve using the original image and PCA-based image occasionally fluctuated in the first half (0–15 epochs), but the overall convergence trend, highest accuracy rate, and the least loss were obvious in the second half (16–30 epochs). This meant that for users who pursue model stability and stable performance, the original image and PCA-based image with SegNet are also feasible choices.

3.5. Prototype

We created a prototype to verify the availability of our system; Figure 6 shows the interface on which users can upload images for recognition using the drag-and-drop function. If either the length or width fails to meet the requirement of being a multiple of 256, adjacent areas are added individually to prevent future predictions that could affect the result. The predicted results after the segmentation and representation of the returned results are presented at the front end after recognition at the back end, providing quick feedback about tree species (Figure 7).

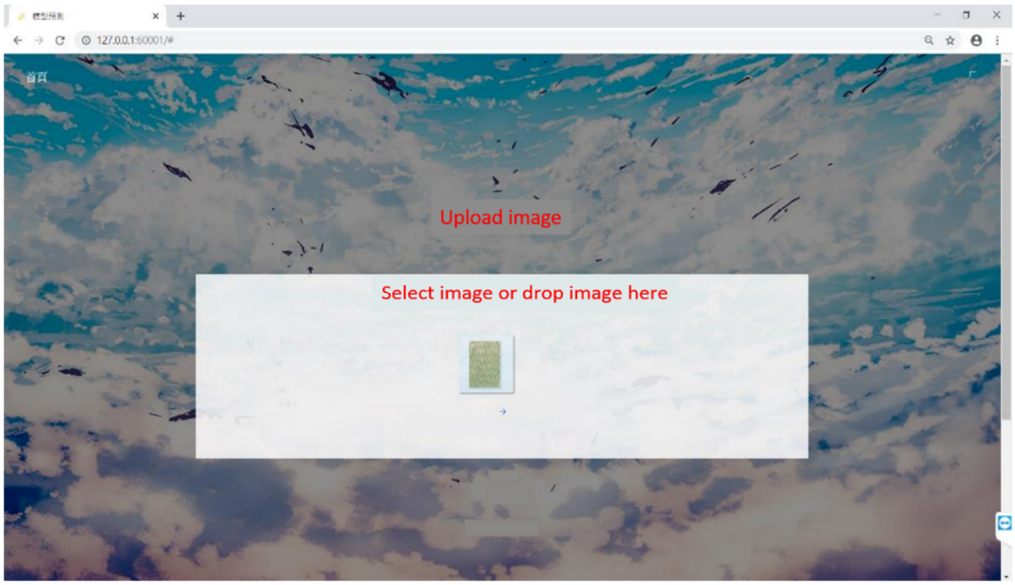


Figure 6. Proposed user interface for uploading images.

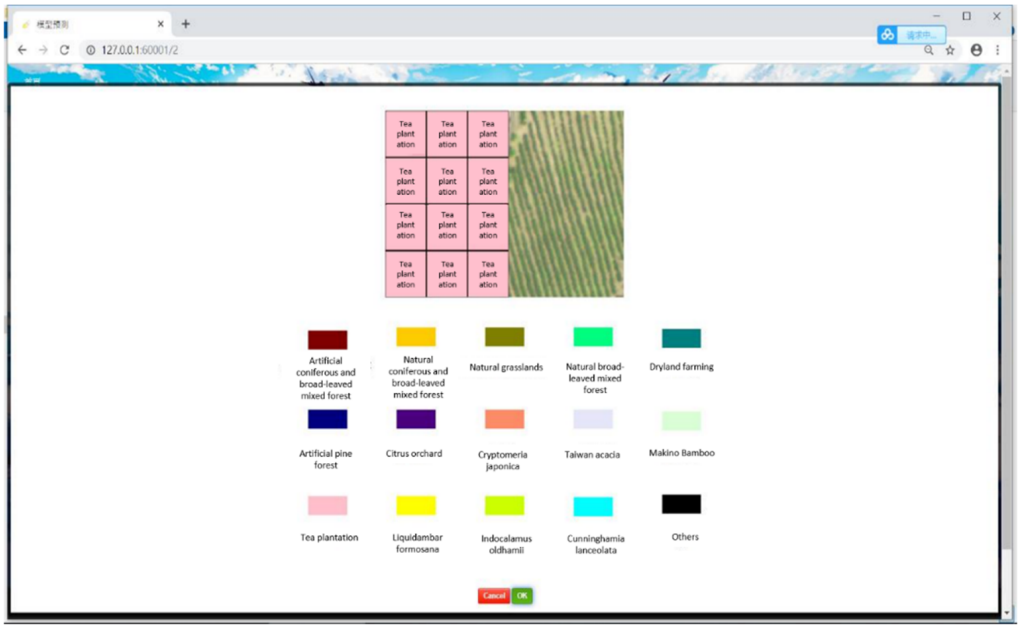


Figure 7. Representation of returned results in a browser.

4. Discussion

4.1. Comprehensive Evaluation of Model Application Efficiency

This research was conducted in the field of environmental resource management. The main technologies include numerical aerial photography technology, field plant survey technology, landscape and space statistical science, and image processing and analysis. We compared the stability, time consumption, and classification accuracy of three models (VGG19, ResNet50, and SegNet) by conducting experiments on two datasets (14 and seven forest and vegetation types) of three different photographs (original aerial photograph, CS-based aerial photograph, and PCA-based aerial photograph).

In terms of classification accuracy, VGG19 outperformed the other two models. When using any given type of image, the classification accuracy decreased in the order VGG19 > SegNet > ResNet50. In addition, the PCA-converted image model had higher accuracy than the CS-based image and the original image models. In Tables 4 and 12, the results for the 14 forest and vegetation types showed that preprocessing PCA-based images effectively improved the model accuracy. For the 14 forest and vegetation types dataset, the model efficiency for using the three types of images decreased in the following order: PCA > CS > original, as revealed in Tables 4, 9 and 12.

The above research results show that CNNs can be applied in an end-to-end manner such that the low-level to high-level features of an input image can be learned without using either manual feature extraction or handcrafted features. However, the object of this research was forest vegetation with fuzzy boundaries and high homogeneity among different types. Therefore, the most important task was to keep sufficient information for classification when inputting training data and reducing noise that either confuses or reduces the classification accuracy at the same time. In addition, the above research results indicate that both PCA and CS effectively improved the classification accuracy, but CS seemed to remove too many useful detailed features for classification (for example, crown shape, canopy curvature, branch characteristics, branches, and leaf growth patterns). Therefore, the effect of CS was less desirable than using PCA conversion. Conversely, PCA used only the principal axis transformation method to enhance the strength of the existing information. It did not subjectively preserve the main tree classification features or perform a large amount of image homogenization as in the CS conversion. Therefore, it was more in line with the calculation characteristics of CNN.

In terms of time consumption, previous research indicated that some models with higher classification accuracy consume more system resources and require more calculation time [41,42]. Take SegNet, which is good at semantic image segmentation, for example. Badrinarayanan's [43] SegNet is better than the other two FCNs (learned deconv) required by Long et al. [44] and the DeepLab-LargeFOV model of Chen et al. [45]. The training time is much longer. Munteanu et al. [46] pointed out that more training parameters means that the training times of SegNet for road detection are much longer than those of U-Net or ResNet based on their experimental results; moreover, ResNet50 took fewer epochs to train than SegNet. When using the smallest number of parameters (Table 3), the training and model inference costs are similarly smaller. SegNet is slower than FCN and DeepLabv1 because SegNet contains the decoder architecture.

Tables 6, 10 and 14 provide a comparison of the calculation times for different models. Compared to the calculation time for the dataset with seven forest and vegetation types, the 14 forest and vegetation types dataset required longer training and testing times when using VGG19 and ResNet50 due to it being larger. However, it required a longer calculation time (including training and testing) than VGG19 or ResNet50 when using SegNet. In our partial code, we used the same weights as ImageNet in VGG19 and Resnet50; however, we did not use ImageNet weights in SegNet because it was another type of network. This may be one reason why the training and testing times of VGG19 and ResNet50 were faster than those of SegNet.

In terms of stability, as seen in Figure 5, which presents the curves of the training and validation losses and accuracies within 30 epochs, the performance of ResNet50 was unstable, SegNet could quickly converge within 10 epochs, and the training and validation losses for VGG19 decreased steadily within 30 epochs. Evidently, when using ResNet50 (which is different from traditional network architectures such as AlexNet, OverFeat, or VGG), its identity mapping layers can prevent the network from degrading when the depth increases. In addition, it introduced global average pooling instead of a fully connected layer to reduce the size of the model. However, the classification objects in this study were forest plantations with a large degree of appearance variation. Identity mapping and residual mapping cannot guarantee its network performance or effective mapping efficiency because all types of trees are intertwined, and there are no clear individual boundaries or fixed and consistent body types. Furthermore, importing residuals does not necessarily have a greater effect than adjusting the weights. These factors might explain why ResNet50 showed less stable performance than VGG19 or SegNet.

We compared our results with previous studies. Bianco et al. presented an in-depth analysis of the majority of the deep neural networks (DNNs) proposed for state-of-the-art image recognition [47]. They reported inference times per image for each DNN model for both the NVIDIA Titan X Pascal GPU and the NVIDIA Jetson TX1. From their results comparing inference time vs. batch size, VGG19 needs more inference time and memory consumption than ResNet50. However, we found the opposite in our case due to smaller parameters in Table 3.

Finally, we concluded that SegNet outperformed VGG19 in terms of model stability and convergence, but it did not perform as well as VGG19 in classification accuracy. Compared to the original images, the PCA conversion images could effectively improve various classification capabilities. Therefore, in the future, it is recommended that original images be pre-processed before being used as inputs.

4.2. The Impact of Model Selection on the Effectiveness of Information for Classification

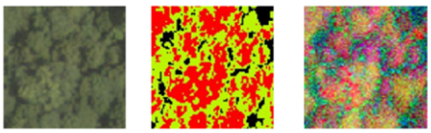
Most front-line staff in forestry management and investigation lack programming capabilities but are quite experienced in aerial photograph data collection and processing. If we can ensure the regularity of aerial photograph pre-processing, model selection, and classification effectiveness or relevance, it will greatly improve the efficiency and accuracy of subsequent forest and vegetation surveys.

If the current CNN-based feature tracking method is applied to large areas and various types of image classification, it will still be affected by several obstacles. Most off-the-shelf CNN models obtain general representations of targets on large-scale image classification datasets. As we all know, different convolutional layers encode different types of features. Features taken from higher convolutional layers retain rich coarse-level semantics but are relatively less able to grasp accurate positioning or scale estimation. Conversely, features extracted from the lower convolutional layer retain more geometric structure to capture the spatial positions of specific targets, which helps to accurately locate target objects. However, if the objects to be classified have a high degree of feature similarity (such as in a plant canopy where most of the shapes are very similar and of various sizes), the higher the number of layers, the easier it is for weight values of different categories to become too close in the final classification weight calculation, resulting in false results. For example, as shown in Figures 8 and 9, this study used VGG19 to train *Cryptomeria Japonica* (Type 8), which belongs to forestland, and tea plantations (Type 11), which belong to farming. The last layers of "conv1" and "conv3" produced larger differences between Types 8 and 11. However, when entering "conv5," as subtle features and common features were successively proposed, the probability of the intermediate hidden layer being activated by ReLU and the sotfmax activation function also increased, which made the weighted sum of different classification categories quite close. As the number of layers increases, the probabilities of feature overlap and of classification weights being similar will increase significantly. This means that if the number of classifications increases, the increase in the number of convolutional layers of the CNN model may not necessarily effectively improve the classification performance. A CNN must rely on the feature space region to represent object-specific details for reinforcement or try to integrate multiple convolutional superior layers to improve the classification accuracy. Therefore, it can clearly explain why the accuracy of the seven classifications is better than that of the 14 classifications. In other words, to obtain better classification results, we must enhance or highlight the non-intersecting characteristics between different categories instead of simplifying the information. Therefore, the positive benefits of using CS data are minimal. Furthermore, if the input data are first converted by PCA, the differences between each grid in the activation layer during the convolution process will increase. The difference between the categories can be improved in the subsequent comparison of the weight difference. Essentially, although the CNN model has the ability to capture deep detailed features, it still needs to rely on feature combination or the large shallow features provided by the lower convolutional layer for advanced classification when facing objects with extremely similar details.

Regarding the interactions between the input data and the CNN model, many previous studies suggested improving the classification accuracy highlighting useful features and suppressing irrelevant redundancies. However, in this research, the subjective elimination of redundant information from aerial photographs (such as CS pre-processing) seemed to cause model drift in classification and even lead to failed target tracking. When the amount of information is limited, if the characteristics of classified objects are similar, it is easy to cause complete commission and omission biases. Particularly in the part of ResNet50, the residual block was originally added to solve the problem that the gradient could not flow to the next layer when the network was gradually deepened. From another perspective, if the age distribution and scale of forests and vegetation vary greatly (for example, when many large tree canopies and small tree canopies exist in close proximity to each other), the residual block will cause scale disorders and dimensionality inconsistencies; this problem can be eliminated by PCA pre-processing. The reason for this is that PCA conversion brings together highly correlated variables through dimensionality reduction, which highlights useful features to a certain extent and reduces the impact of redundant information. Therefore, no matter whether we use VGG19, ResNet50, or SegNet, we can obtain good results by inputting the PCA-converted layer.

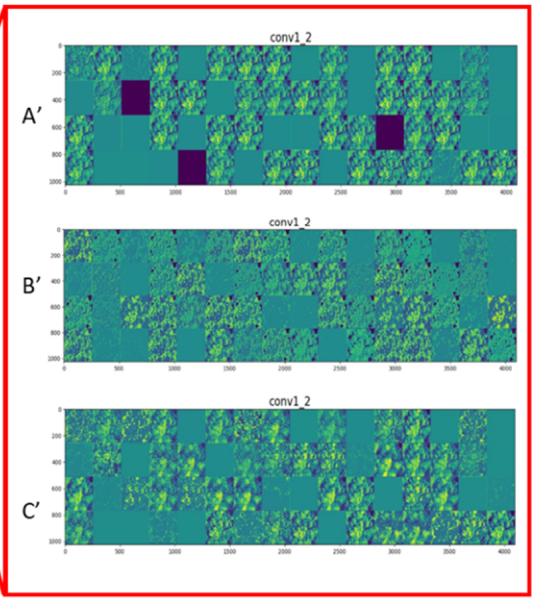
VGG19

No.8



A B C

Layer name	#Filters	#Parameters	#Activations
input			150K
conv1_1	64	1.7K	3.2M
conv1_2	64	36K	3.2M
max pooling			802K
conv2_1	128	73K	1.6M
conv2_2	128	147K	1.6M
max pooling			401K
conv3_1	256	300K	802K
conv3_2	256	600K	802K
conv3_3	256	600K	802K
conv3_4	256	600K	802K
max pooling			200K
conv4_1	512	1.1M	401K
conv4_2	512	2.3M	401K
conv4_3	512	2.3M	401K
conv4_4	512	2.3M	401K
max pooling			100K
conv5_1	512	2.3M	100K
conv5_2	512	2.3M	100K
conv5_3	512	2.3M	100K
conv5_4	512	2.3M	100K
max pooling			25K
fc6		103M	4K
fc7		17M	4K
output		4M	1K



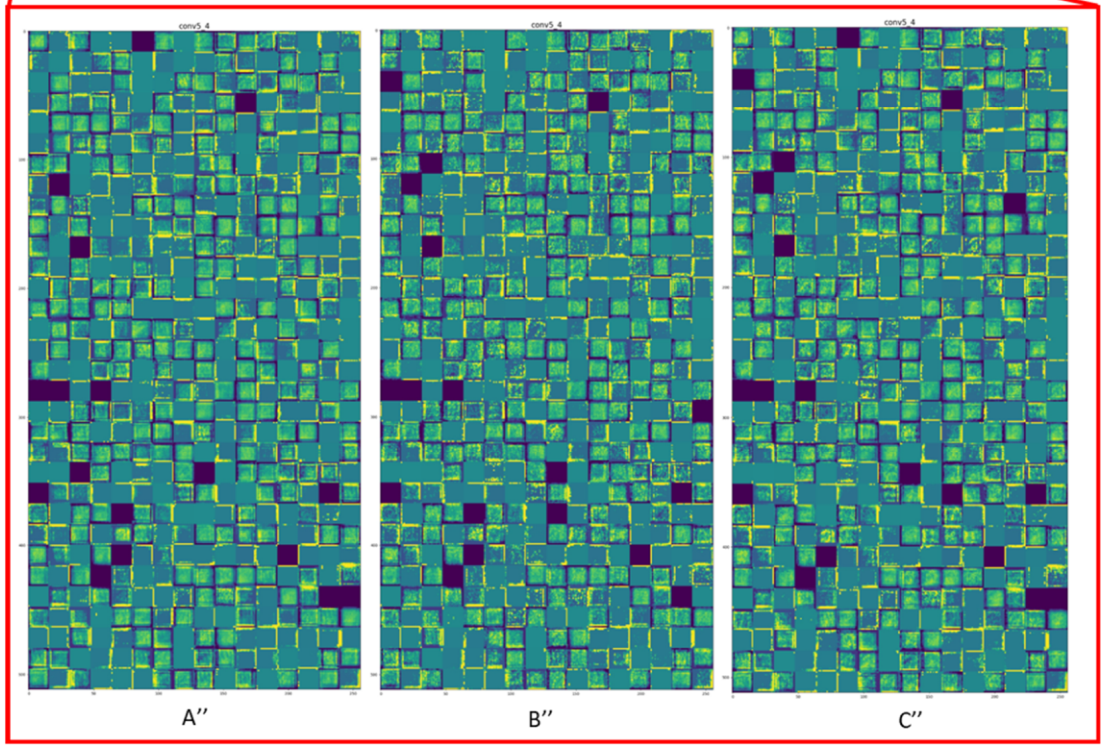


Figure 8. Visualization of feature channels for three different kinds of input in the last layer of the "conv1" and "conv5" stages in VGG19 (vegetation type: *Cryptomeria Japonica* (Type 8)).

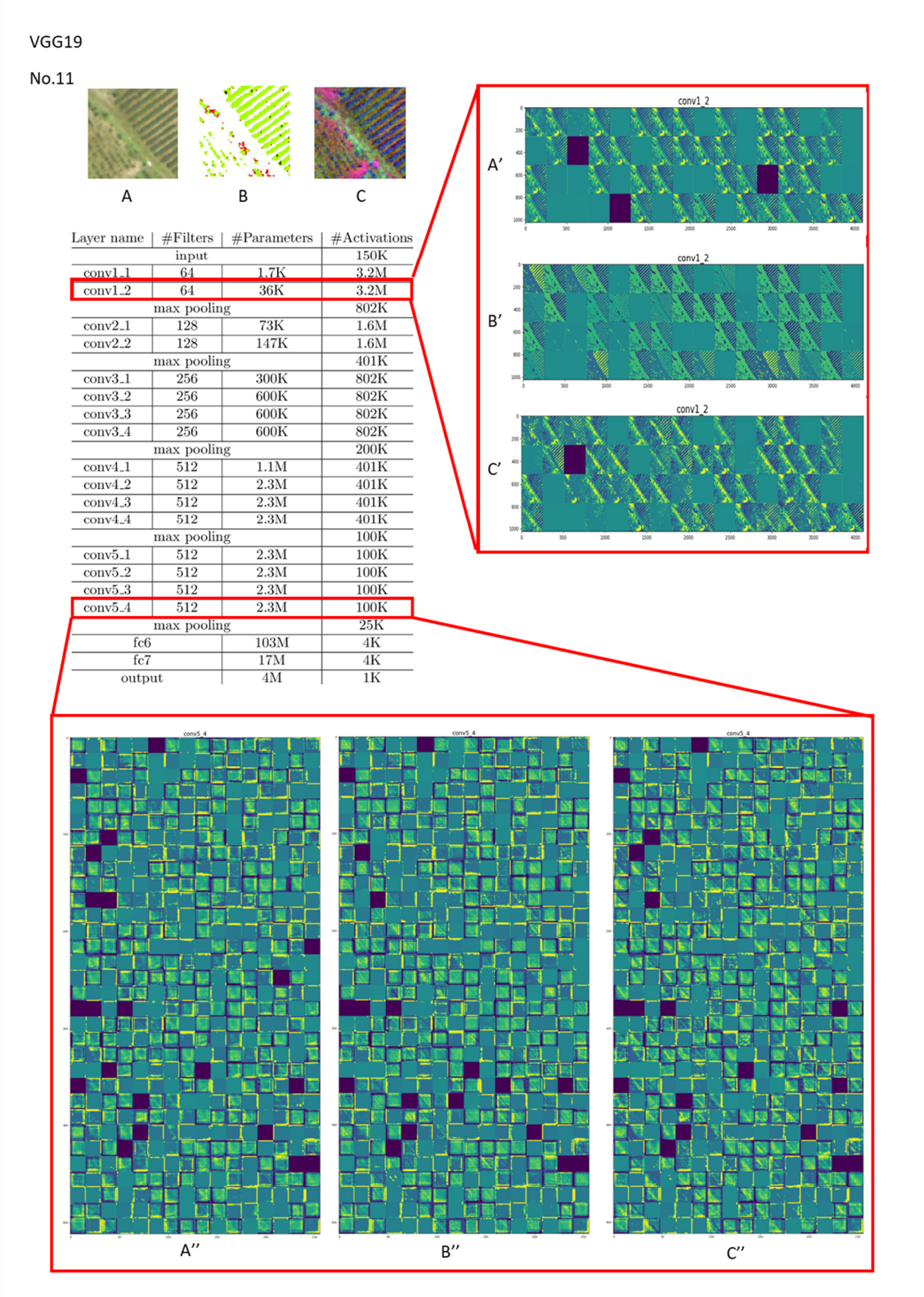


Figure 9. Visualization of feature channels for three different kinds of input in the last layer of the "conv1" and "conv5" stages in VGG19 (vegetation type: tea plantation (Type 11)).

This study also found that the simultaneous comparison of feature similarities between frames and within frames could effectively improve the accuracy of classification. Nevertheless, if the step size is set too small to pursue the similarity between frames, it

will significantly increase the computing time. VGG19 and ResNet50 used in this study ignored the inter-frame attention in the tracker design and only used the original depth feature frame obtained from the first frame to match the candidate patches in the following frames. This just emphasized the degree of intra-frame overlap and made the model application easier.

In Figure 5, we also found that SegNet could quickly converge during training and validation and achieved 100% accuracy when combined with aerial photographs processed by PCA. The reason for this is due to both the intensity of the available information in the PCA pre-processing concentration and the important reason that SegNet maintains translation invariance when performing small spatial displacements on input images. Since forests or vegetation growing on the hillsides of Taipei are all regional, they have the characteristics of group gatherings in spatial distribution. Therefore, if the location characteristics are marked in the calculation process of feature comparison, and the appropriate pooling operation and activation function are used, this can greatly increase the classification speed and result in a faster convergence curve. However, the result map would also show more complete segments, which is why SegNet with PCA layers can achieve the best combination effect.

5. Conclusions

In this study, the interpretation and classification of forest cover were performed for forest areas in Taipei City using numerous aerial photographs and three CNN calculation models. Verification was performed using actual ground data. The results revealed that the original non-preprocessed inputs were unsuitable for forestland classification due to the high complexity of forest characteristics.

Additionally, CNN classification calculations were found to be unsuitable because of the large coverage and high similarity of forest features. The performance of classification calculations when using excessively simplified content on forest feature layers were also poor. The combination of the PCA algorithm extracted using a reduction in the number of dimensions and the SegNet algorithm resulted in the highest CNN extraction capacity and achieved a high degree of simulation stability after a certain number of training cycles; thus, this combination is suggested by the present study.

We proposed a real and usable forest and vegetation type identification system based on the study contributions. In image analysis, we discovered that when using the same model, images subjected to PCA or feature processing produced higher identification accuracy compared to unprocessed data. Moreover, the computational speed was higher when using seven forest and vegetation types rather than 14 forest and vegetation types. In future studies, SegNet should be used if the wait time is not a major concern. However, VGG19 has the highest overall speed performance.

Author Contributions: F.-C.L. and Y.-C.C. conceived and designed the experiments; F.-C.L. performed the experiments; Y.-C.C. and F.-C.L. analyzed the data and contributed to the materials and analysis tools; Y.-C.C. wrote the manuscript. Finally, Y.-C.C. and F.-C.L. provided relevant materials and valuable comments for this paper. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the Ministry of Science and Technology, Taiwan (Grant Number: MOST 107-2410-H-035-032 and MOST 110-2121-M-035-001).

Data Availability Statement: Scripts and associated data are available as URL link presented in the article.

Acknowledgments: The authors thank all participants for their contributions and project funding.

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

References

1. Ardila, J.P.; Bijker, W.; Tolpekin, V.A.; Stein, A. Context-sensitive extraction of tree crown objects in urban areas using VHR satellite images. *Int. J. Appl. Earth Obs. Geoinf.* **2012**, *15*, 57–69. [CrossRef]
2. Chuang, Y.C.M.; Shiu, Y.S. A comparative analysis of machine learning with WorldView-2 pan-sharpened imagery for tea crops mapping. *Sensors* **2016**, *16*, 594. [CrossRef]
3. Kasischke, E.S.; French, N.H.; Harrell, P.; Christensen, N.L.; Ustin, S.L.; Barry, D. Monitoring of wildfires in boreal forests using large area AVHRR NDVI composite image data. *Remote Sens. Environ.* **1993**, *45*, 61–71. [CrossRef]
4. Remmel, T.K.; Perera, A.H. Fire mapping in a northern boreal forest: Assessing AVHRR/NDVI methods of change detection. *For. Ecol. Manag.* **2001**, *152*, 119–129. [CrossRef]
5. Rudol, P.; Doherty, P. Human Body Detection and Geolocalization for UAV Search and Rescue Missions Using Color and Thermal Imagery. In Proceedings of the IEEE conference on Aerospace, Big Sky, MT, USA, 1–8 March 2008. [CrossRef]
6. Sothe, C.; De Almeida, C.M.; Schimalski, M.B.; Liesenberg, V.; La Rosa, L.E.; Castro, J.D.; Feitosa, R.Q. A comparison of machine and deep-learning algorithms applied to multisource data for a subtropical forest area classification. *Int. J. Remote Sens.* **2020**, *41*, 21943–21969. [CrossRef]
7. Hou, J.; Si, Y.; Yu, X. A Novel and Effective Image Super-Resolution Reconstruction Technique via Fast Global and Local Residual Learning Model. *Appl. Sci.* **2020**, *10*, 1856. [CrossRef]
8. Yun, J.H.; Lim, H.J. Image Restoration Using Fixed-Point-Like Methods for New TVL1 Variational Problems. *Electronics* **2020**, *9*, 735. [CrossRef]
9. Han, D.; Liu, Q.; Fan, W. A new image classification method using CNN transfer learning and web data augmentation. *Expert Syst. Appl.* **2018**, *95*, 243–256. [CrossRef]
10. Kamilaris, A.; Prenafeta-Boldú, F.X. Deep learning in agriculture. *Surv. Comput. Electron. Agric.* **2018**, *147*, 70–90. [CrossRef]
11. Mohammadimanesh, F.; Salehi, B.; Mahdianpari, M.; Gill, E.; Molinier, M. A new fully convolutional neural network for semantic segmentation of polarimetric SAR imagery in complex land cover ecosystem. *ISPRS J. Photogramm. Remote Sens.* **2019**, *151*, 223–236. [CrossRef]
12. Traore, B.B.; Kamsu-Foguem, B.; Tangara, F. Deep convolution neural network for image recognition. *Ecol. Inform.* **2018**, *48*, 257–268. [CrossRef]
13. Haralick, R.M.; Shaunmmugam, K.; Dinstein, I. Textural Features for Image Classification. *IEEE Trans. Syst. Man Cybern.* **1973**, *67*, 786–804. [CrossRef]
14. Duque, J.C.; Patino, J.E.; Ruiz, L.A.; Pardo-Pascual, J.E. Measuring intra-urban poverty using land cover and texture metrics derived from remote sensing data. *Landsc. Urban. Plan.* **2015**, *135*, 11–21. [CrossRef]
15. Leichtle, T.; Geiß, C.; Wurm, M.; Lakes, T.; Taubenböck, H. Unsupervised change detection in VHR remote sensing imagery—An object-based clustering approach in a dynamic urban environment. *Int. J. Appl. Earth Obs. Geoinf.* **2017**, *54*, 15–27. [CrossRef]
16. Wang, W.; Yang, N.; Zhang, Y.; Wang, F.; Cao, T.; Eklund, P. A review of road extraction from remote sensing images. *J. Traffic Transp. Eng.* **2016**, *3*, 271–282. [CrossRef]
17. Schmidhuber, J. Deep learning in neural networks: An overview. *Neural Netw.* **2015**, *61*, 85–117. [CrossRef] [PubMed]
18. Hubel, D.; Wiesel, T. Receptive fields and functional architecture of monkey striate cortex. *J. Physiol.* **1968**, *195*, 215–243. [CrossRef] [PubMed]
19. Cheng, G.; Han, J. A survey on object detection in optical remote sensing images. *ISPRS J. Photogramm. Remote Sens.* **2016**, *117*, 11–28. [CrossRef]
20. Grinias, I.; Panagiotakis, C.; Tziritas, G. MRF-based segmentation and unsupervised classification for building and road detection in peri-urban areas of high-resolution satellite images. *ISPRS J. Photogramm. Remote Sens.* **2016**, *122*, 145–166. [CrossRef]
21. Zhang, P.; Gong, M.; Su, L.; Liu, J.; Li, Z. Change detection based on deep feature representation and mapping transformation for multi-spatial-resolution remote sensing images. *ISPRS J. Photogramm. Remote Sens.* **2016**, *116*, 24–41. [CrossRef]
22. Hinton, G.E.; Salakhutdinov, R.R. Reducing the dimensionality of data with neural networks. *Science* **2016**, *313*, 504–507. [CrossRef]
23. Glorot, X.; Bordes, A.; Bengio, Y. Deep sparse rectifier networks. *AISTATS* **2011**, *15*, 315–323.
24. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. ImageNet classification with deep convolutional neural networks. *Commun. ACM* **2017**, *60*, 84–90. [CrossRef]
25. Simonyan, K.; Zisserman, A. Very deep convolutional networks for large-scale image recognition. *arXiv* **2014**, arXiv:1409.1556.
26. 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.
27. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778. [CrossRef]
28. Huang, G.; Liu, Z.; Weinberger, K.Q. Densely connected convolutional networks. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 2261–2269. [CrossRef]
29. Kwan, C.; Gribben, D.; Ayhan, B.; Li, J.; Bernabe, S.; Plaza, A. An Accurate Vegetation and Non-Vegetation Differentiation Approach Based on Land Cover Classification. *Remote Sens.* **2020**, *12*, 3880. [CrossRef]
30. Kwan, C.; Ayhan, B.; Budavari, B.; Lu, Y.; Perez, D.; Li, J.; Bernabe, S.; Plaza, A. Deep Learning for Land Cover Classification Using Only a Few Bands. *Remote Sens.* **2020**, *12*, 2000. [CrossRef]

31. Ayhan, B.; Kwan, C.; Budavari, B.; Kwan, L.; Lu, Y.; Perez, D.; Li, J.; Skarlatos, D.; Vlachos, M. Vegetation Detection Using Deep Learning and Conventional Methods. *Remote Sens.* **2020**, *12*, 2502. [CrossRef]
32. Li, M.; Stein, A. Mapping Land Use from High Resolution Satellite Images by Exploiting the Spatial Arrangement of Land Cover Objects. *Remote Sens.* **2020**, *12*, 4158. [CrossRef]
33. Geng, J.; Wang, H.; Fan, J.; Ma, X. Deep supervised and contractive neural network for sar image classification. *IEEE Trans. Geosci. Remote. Sens.* **2017**, *55*, 2442–2459. [CrossRef]
34. Hartling, S.; Sagan, V.; Sidike, P.; Maimaitijiang, M.; Carron, J. Urban tree species classification using a worldview-2/3 and liDAR data fusion approach and deep learning. *Sensors* **2019**, *19*, 1284. [CrossRef]
35. Trichon, V. Crown typology and the identification of rain forest trees on large-scale aerial photographs. *Plant. Ecol.* **2001**, *153*, 301–312. [CrossRef]
36. Kumar, N.; Sukavanam, N. Detecting Helmet of Bike Riders in Outdoor Video Sequences for Road Traffic Accidental Avoidance. In Proceedings of 18th International Conference on Intelligent Systems Design and Applications (ISDA 2018), Vellore, India, 6–8 December 2018; Volume 2, pp. 24–33.
37. Sivagami, R.; Srihari, J.; Ravichandran, K.S. Analysis of Encoder-Decoder Based Deep Learning Architectures for Semantic Segmentation in Remote Sensing Images. In Proceedings of 18th International Conference on Intelligent Systems Design and Applications (ISDA 2018), Vellore, India, 6–8 December 2018; Volume 2, pp. 332–341.
38. Veeraballi, R.K.; Nagugari, M.S.; Annavarapu, C.S.R.; Gownipuram, E.V. Deep Learning Based Approach for Classification and Detection of Papaya Leaf Diseases. In Proceedings of 18th International Conference on Intelligent Systems Design and Applications (ISDA 2018), Vellore, India, 6–8 December 2018; Volume 1, pp. 291–302.
39. Natesan, S.; Armenakis, C.; Vepakomma, U. Resnet-based tree species classification using uav images. In Proceedings of the 2019 International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Enschede, The Netherlands, 10–14 June 2019.
40. Srivastava, N.; Hinton, G.; Krizhevsky, A.; Sutskever, I.; Salakhutdinov, R. Dropout: A simple way to prevent neural networks from overfitting. *J. Mach. Learn. Res. (JMLR)* **2014**, *15*, 1929–1958.
41. Padalkar, G.R.; Khambete, M.B. Analysis of Basic-SegNet Architecture with Variations in Training Options, Springer Nature Switzerland AG 2020. In Proceedings of 18th International Conference on Intelligent Systems Design and Applications (ISDA 2018), Vellore, India, 6–8 December 2018; Volume 1, pp. 727–735.
42. Teso-Fz-Betoño, D.; Zulueta, E.; Sánchez-Chica, A.; Fernandez-Gamiz, U.; Saenz-Aguirre, A. Semantic Segmentation to Develop an Indoor Navigation System for an Autonomous Mobile Robot. *Mathematics* **2020**, *8*, 855. [CrossRef]
43. Badrinarayanan, V.; Kendall, A.; Cipolla, R. SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *39*, 2481–2495. [CrossRef] [PubMed]
44. Long, J.; Shelhamer, E.; Darrell, T. Fully convolutional networks for semantic segmentation. In Proceedings of 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, USA, 7–12 June 2015; Volume 1, pp. 3431–3440.
45. Chen, L.C.; Papandreou, G.; Kokkinos, I.; Murphy, K.; Yuille, A. DeepLab: Semantic image segmentation with deep convolutional nets and fully connected CRFs. In Proceedings of 3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, 7–9 May 2015.
46. Munteanu, A.; Selea, T.; Neagul, M. Deep Learning Techniques Applied for Road Segmentation. In Proceedings of the 21st International Symposium on Symbolic and Numeric Algorithms for Scientific Computing (SYNASC), Timisoara, Romania, 4–7 September 2019; pp. 297–303.
47. Bianco, S.; Cadène, R.; Celona, L.; Napoletano, P. Benchmark analysis of representative deep neural network architectures. *IEEE Access* **2018**, *6*, 64270–64277. [CrossRef]