Mühendislik Bilimleri ve Tasarım Dergisi
12(2), 328-344, 2024
e-ISSN: 1308-6693

Journal of Engineering Sciences and Design
DOI: 10.21923/jesd.1372788

Araştırma Makalesi Research Article

INCEPTION SH: A NEW CNN MODEL BASED ON INCEPTION MODULE FOR CLASSIFYING SCENE IMAGES

Sedat METLEK[1]*, Halit ÇETİNER[2]

[1] Burdur Mehmet Akif Ersoy University, Vocational School of Technical Sciences, Burdur, Türkiye
[2] Isparta University of Applied Sciences, Vocational School of Technical Sciences, Isparta, Türkiye

Keywords

Deep Learning,
Computer Vision,
CNN,
Scene Classification,
UAV.

Abstract

In this study, a light-weight model with an optimum block structure that can be used in autonomous unmanned aerial vehicles (UAVs) was designed. The Inception SH model, which was developed based on the Inception V3 model, was compared on "Intel Image Dataset", a publicly available dataset in the literature. As a result of the comparison, values of 0.882, 0.883, 0.882 and 0.882 were obtained for the accuracy, precision, recall, and F1 score metrics for the Inception V3 model, respectively. In the Inception SH model, values of 0.958, 0.957, 0.974 and 0.967 were obtained for accuracy, precision, recall and F1 score metrics, respectively. As can be seen from these values, the proposed Inception SH model offers higher performance values than the underlying Inception V3 model. The Inception SH model was compared with different models in the literature using the same data set and was superior in accuracy, precision, recall and F1 score metrics compared to the compared models. According to the results obtained, it is predicted that the Inception SH model can be used as a lightweight model in various IoT devices, considering the popularity of autonomous UAVs.

INCEPTION SH: SAHNE GÖRÜNTÜLERININ SINIFLANDIRILMASINDA INCEPTION MODÜL TABANLI YENI BIR CNN MODELI

Anahtar Kelimeler

Derin Öğrenme,
Bilgisayarla Görü,
CNN,
Sahne Sınıflandırma,
IHA.

Öz

Bu çalışmada otonom insansız hava araçlarında (İHA) kullanılabilecek optimum seviyede blok yapısına sahip hafif ağırlıklı bir model tasarlanmıştır. Inception V3 modeli temel alınarak geliştirilen Inception SH modeli, literatürde halka açık bir veri seti olan "Intel Image Dataset" üzerinde karşılaştırılmıştır. Karşılaştırma sonucunda Inception V3 modeli için doğruluk, kesinlik, geri çağırma ve F1 skoru metrikleri için sırasıyla 0,882, 0,883, 0,882 ve 0,882 değerleri elde edilmiştir. Inception SH modelinde ise doğruluk, kesinlik, geri çağırma ve F1 skoru metrikleri için sırasıyla 0,958, 0,957, 0,974 ve 0,967 değerleri elde edilmiştir. Bu değerlerden de anlaşılacağı üzere, önerilen Inception SH modeli, temel alınan Inception V3 modeline göre daha yüksek performans değerleri sunmaktadır. Inception SH modeli aynı veri setini kullanan literatürdeki farklı modellerle de karşılaştırılmış ve karşılaştırılan modellere göre doğruluk, kesinlik, geri çağırma ve F1 skoru metriklerinde üstünlük sağlamıştır. Elde edilen sonuçlara göre, otonom İHA'ların popülerliği de göz önünde bulundurulduğunda, Inception SH modelinin çeşitli IoT cihazlarında hafif bir model olarak kullanılabileceği öngörülmektedir.

Alıntı / Cite

Metlek, S., Çetiner, H., (2024). Inception SH: A New Inception-Based CNN Model for Classification of Scene Images, Journal of Engineering Sciences and Design, 12(2), 328-344.

Yazar Kimliği / Author ID (ORCID Number)	Makale Süreci / Article Process	
S. Metlek, 0000-0002-0393-9908	Başvuru Tarihi / Submission Date	08.10.2023
H. Çetiner, 0000-0001-7794-2555	Revizyon Tarihi / Revision Date	14.04.2024
	Kabul Tarihi / Accepted Date	07.07.2024
	Yayım Tarihi / Published Date	30.06.2024

* İlgili yazar / Corresponding author: sedatmetlek@mehmetakif.edu.tr, +90-248-213-4580

INCEPTION SH: A NEW CNN MODEL BASED ON INCEPTION MODULE FOR CLASSIFYING SCENE IMAGES

Sedat METLEK[1†] Halit ÇETİNER[2]

[1] Burdur Mehmet Akif Ersoy University, Vocational School of Technical Sciences, Burdur, Türkiye
[2] Isparta University of Applied Sciences, Vocational School of Technical Sciences, Isparta, Türkiye

Highlights

- Inception SH, a new Inception-based deep learning model that automatically detects and classifies the environment of an autonomous UAV is proposed.
- A new thirteen-step deep learning model is proposed without using any lightweight pre-trained architectural model that can run on embedded systems.
- According to the accuracy, recall, precision, and F1 score metrics commonly used in the literature, the Inception SH model provides a better result by approximately 4% on the same data set.

Graphical Abstract

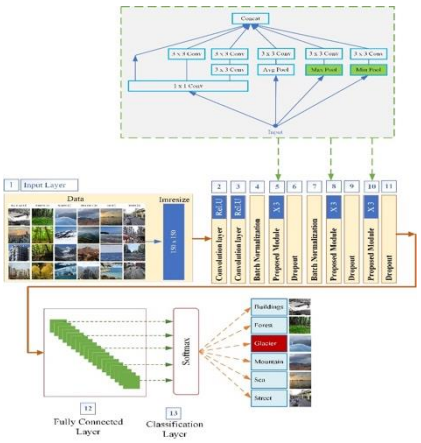


Figure. Inception SH model

Purpose and Scope

The purpose of the study is to develop a deep learning algorithm that allows UAVs to autonomously evaluate their environment and make decisions as a result. Another purpose of the study is that the developed architecture is lightweight and can work in many embedded systems.

Design/methodology/approach

The proposed Inception SH architecture was created with a CNN-based approach by developing Inception modules. In parallel with this, average, max, and min pooling operations were carried out by taking the input image directly. At this stage, the images were transferred directly to these filters and the [1×1] convolution process was not applied.

Findings

Inception SH model provided a superior performance result compared to studies performed using the same dataset. While the proposed model provides an average performance of 95%, the closest study in the literature achieved a performance value of 91%. It is obvious that the proposed model can be used on different platforms because it is lighter than many models in the literature. Considering the popularity of autonomous UAVs, it is anticipated that the Inception SH model, developed as a lightweight model, can also be used in IoT devices.

Originality

In this study, a deep learning model based on Inception V3 was developed that will enable UAVs to make instant decisions about the environment autonomously, independently of the operators using the images. Although the proposed system is based on the Inception V3 model, the performance difference is presented in detail. A comparison was made with other studies in the literature using the same dataset.

† Corresponding author: sedatmetlek@mehmetakif.edu.tr, +90-248-213-4580

1. Introduction

Today, Unmanned Aerial Vehicles (UAVs) are used in a wide variety of applications, as they have proven to be both manned and autonomous in a variety of environments and missions (Yuan, Liu, & Zhang 2015;Menouar et al. 2017). As a result, it is known that UAV production has increased worldwide, as can be seen from the reports produced on the subject (Finnegan 2017; Grand View 2023). Especially when evaluated in terms of costs, it is seen that they are cost-effective systems compared to aircraft, satellite remote sensing systems, and other air vehicles (Matese et al. 2015).

When the studies conducted with UAVs are examined in general, it is seen that many different detection sensors coexist on UAVs, and they are used synchronously at the same time (Amarasingam et al. 2022). For example, it is seen that agricultural products can be classified with high accuracy with opto-electro and thermal cameras placed on the UAV (Shahi et al. 2023). But the important point here is that, unlike cameras, many sensors are used at the same time for accurate location and graduation. The first of these sensors that come to mind are acoustic sensors. Acoustic sensors are generally resistant to environmental conditions. But their limited effective range is extremely limited. They are therefore less widely used than cameras. In addition to these sensors, radar sensors are also used, which provide precise localization and have a much longer effective range than acoustic sensors. These sensors are not affected by environmental conditions like acoustic sensors.

The operators of the UAVs make instant decisions based on the data obtained from these sensors and direct the UAVs. In addition to the advantage of having many sensors on UAVs, there are also some disadvantages. Because information from many different sensors must be combined to produce a result. In many UAV systems today, warning signals are received from acoustic sensors and radar data, and maneuvers are performed by the operator after verification with images from the camera. This is the most important obstacle in the development of autonomous UAVs. Because while the information received from many different sensors can be evaluated clearly, this situation needs to be confirmed through the image taken from the cameras.

When an obstacle appears in front of the UAV, the acoustic sensor can detect the obstacle, but not what it is. For example, a very high antenna, a building or a mountain in front of the UAV cannot be clarified with the information received from the acoustic sensor. In such cases, it is essential to confirm what the obstacle is with the view from the cameras. Analysis of UAVs shows that extremely high-resolution images can be obtained. These images are usually sent immediately to the ground station, where they are analyzed by operators. Deep learning methods can be actively used to classify such images, even at a basic level. This is because it is a popular method that is also actively used in many other applications (Akbay 2022; Çetiner and Metlek 2023; S Metlek and Çetiner 2023; Şenel and Şenel 2022; Tokmak 2022).

When examined in the literature, it is seen that such image analyses are increasing. In some of them, Zeggada and Melgani (Zeggada and Melgani 2017), and Moranduzzo et al. (Moranduzzo et al. 2015) developed a multi-label classification method for unmanned aerial vehicles using images from urban areas. The focus of this work is to develop a classification algorithm that allows a UAV to make inferences about its environment from instantaneous scene images captured by the camera instead of multi-label classification. For this purpose, a recent dataset containing different scene information from six different environments was used. Since the system is intended to be used in an autonomous system, an architectural model that can work on an embedded system has been developed. For this purpose, a new classification model has been developed based on the CNN architecture, which can also run on embedded systems in the basic literature.

The theory behind the proposed approach is to extract the distinctive features that define the basic characteristics of six different natural environments from the scene images and classify them with high accuracy. In this way, false object detections due to misperception of the scene can be avoided in future studies. For example, the aim is to eliminate illogical classification situations, such as encountering a building in a marine environment, so that the UAV can make more accurate decisions autonomously. In this study, a comprehensive study of scene classification has been carried out, which is the crucial point of the study. Researchers favoring this approach will also benefit in terms of resource management, such as time and memory (Grand View 2023).

The main contributions of this article to the literature include the following.

- Inception SH, a new Inception-based deep learning model that automatically detects and classifies the environment of the autonomous UAV, is proposed.
- A new deep learning model consisting of thirteen steps without using any light-weight pre-trained architectural model that can run on embedded systems is proposed.
- A new block structure is proposed based on Inception block structures.

- According to the accuracy, recall, precision, and F1 score metrics commonly used in the literature, the Inception SH model provides a better result by approximately 4% on the same dataset.

The following sections of the article are organized as follows. In the second section, studies in the literature on the subject are presented. In the third section, detailed information about the materials and methods used are given. In the fourth section, the performance results obtained using the proposed CNN model are presented in detail. In the last section, general evaluations are made in the light of the applied approach and the results obtained, and predictions for future studies are presented.

2. Related Works

The main point of focus in the study is that the UAV can classify its environment based on the images obtained from the camera. In the literature, this process is called scene classification. Scene classification is the general name for categorizing scenes in images. Unlike classical classification, scene classification is performed based on the objects in the background of the image. Low-level features have been used in scene classification for years, and some results can be obtained with these features. But despite years of progress, most approaches still fall short of performing at a level appropriate to a variety of real-world environments. Especially when this process is applied to UAVs with very high financial value, any negativity that may occur can cause a large amount of financial loss. There are also difficulties arising from the nature of the work (Huang, Pedoeem, and Chen 2018). However, regardless of the reason, increasing the performance level of scene classification is an essential issue. As a result of the widespread use of high-resolution satellite images, there have been advances in studies on solving the scene classification problem. Studies conducted in recent years in the literature to increase the performance level are briefly summarized below.

Shabbir et al. used a pre-trained CNN architecture called ResNet50 to assign class labels according to image contents (Shabbir et al. 2021). In the ResNet 50 architecture, they carried out automatic classification by making fine adjustments according to the class outputs of the dataset they used. Zou et al. developed a method that selects distinctive features in order to improve the success rate in classifying 2800 scene images consisting of seven categories (Zou et al. 2015). Tuia et al., investigated, tested, and compared the image scene classification method in detail with three active learning models in their work. In their study, they also shared some guidelines for choosing good classification methods for inexperienced users (Tuia et al. 2011). GóChova et al. investigated multi-modal remote sensing image classification. In his work, he summarized the leading popular algorithms used for scene classification. It appears that most of these algorithms are CNN-based deep learning models (Gómez-Chova et al. 2015). Maulik et al. conducted a detailed review of algorithms based on support vector machines (SVM) and semi-supervised SVMs for image scene classification (Maulik and Chakraborty 2017). Li and et al. investigated pixel-level, subpixel-level, and object-based image classification methods in their work and revealed the contribution of spatial-contextual information to image scene classification (Li et al. 2014). Penatti et al. tested the generalization power of deep features (ConvNets) obtained from convolution in two different scenarios for airborne and remote sensing. They found that while ConvNets offer the best performance values for aerial images, they give lower results for remote sensing (Penatti, Nogueira, and Santos 2015). In their study, Hu et al. evaluate the success of pre-trained CNN-based models for high-resolution scene classification. For this, they used two publicly available datasets. They claim that the image features obtained by the two different scenarios they propose offer remarkable performance even with a simple linear classifier (Hu et al. 2015). Wu et al. classified the dataset they used in their study with transfer learning-based architectures called both Inception V3 and Xception. As a result of the classification, they found that the Xception architecture provides superior performance than the Inception V3 architecture. As a result of the study, the effectiveness of the Xception architecture in scene classification has been proven (Wu et al. 2020). To leverage the power of convolutional neural networks in scene classification, Nogueira et al. separately evaluated three strategies: full training, fine-tuning, and using ConvNets as feature extractors. As a result of the evaluation, they found that fine-tuning tends to be the best-performing strategy. They claim that especially the use of features obtained with fine-grained ConvNets with linear SVM will provide very high performance (Nogueira, Penatti, & Dos Santos 2017). Zhang et al. have recently examined deep learning-based approaches used in scene classification. Although they state in their studies that deep learning-based approaches provide excellent performance, they still emphasize that they need to be improved (Zhang, Zhang, and Du 2016). Xia et al. have developed a new classification method called AID for aerial image classification. In their study, they presented the scene classification methods available before 2017 in detail (Xia et al. 2017).

As seen in the literature research, it is seen that the majority of the studies carried out in recent years to increase the performance level of scene classification are CNN-based deep learning studies. This confirms that choosing CNN-based as the basis for the study is the correct approach.

3. Material and Methods

3.1. Material

Today, image classification applications are actively applied in a wide range of applications, from geographic information systems to medical studies. In this study, the focus is on the automatic classification of real-life natural objects in order to build the necessary infrastructure that will enable UAVs to move autonomously. For this reason, the dataset called "Intel Image Classification", which is a publicly available dataset on the subject in the literature, was used. In this dataset, there are 6 classes of natural landscape images: buildings, forest, glacier, mountain, sea and street. These images used in the study are numbered between 0-5 in the order in which they are presented. These images are 150x150 images in RGB format. In the "Intel Image Classification" dataset, there is also a prediction class separate from the training and test classes. This class is not included in the study to avoid ambiguity. The distribution and total number of classes in the dataset used in the study are presented in detail in Table 1.

Table 1. Detailed distributions of the classes in the dataset used in the study

	Buildings	Forest	Glacier	Mountain	Sea	Street
Train	2191	2271	2404	2512	2274	2382
Test	437	474	553	525	510	501
Sum	2628	2745	2957	3037	2784	2883

As presented in Table 1, a total of 16509 images were used in the study. Sample images of the classes in the dataset are also shown in Figure 1. When these images are examined, they are quite similar to the images that can be taken by any UAV. This dataset was preferred in the study due to this similarity.

Figure 1. Samples of Intel Image Classification dataset

3.2. Method

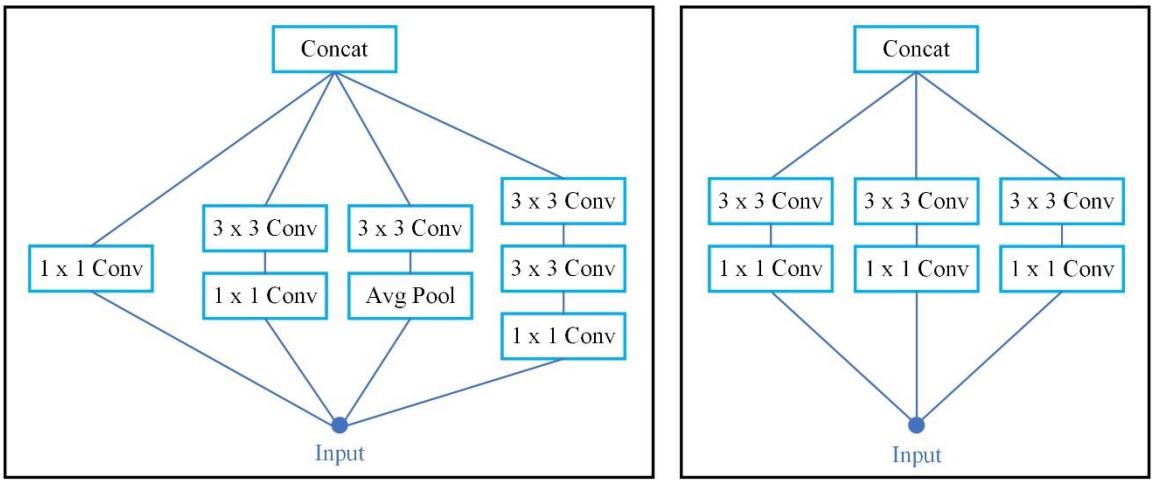
When the literature is examined in general, Machine Learning (ML) and Deep Learning (DL) algorithms are algorithms that show high performance in object detection and classification applications. For this reason, they are still popular in the literature. Machine learning methods can be defined as systems where feature methods such as Histogram of Gradient (HOG) approach, wavelet transforms with detail coefficients, Gray Level Co-occurrence Matrices, Principal Component Analysis, Linear Discriminant Analysis are used as hybrid systems (Noble 2006; Quinlan 1986). In these methods, shallow methods are generally used for feature extraction, but basic machine learning methods are used for classification.

While this approach gives good results on carefully prepared datasets, it performs poorly on complex images with numerous objects or with low similarity within groups. Using basic machine learning methods for feature extraction can lead to unnecessary expenditure of training resources. For this reason, most of the deep learning

approaches in image processing applications are based on the architecture of Convolutional Neural Networks (CNNs). In CNN architecture, features are automatically obtained by applying different filters to the input image successively. The number of features obtained is reduced with the Pooling layer and reduced to certain ranges with activation functions. In the Fully Connected layer, which is the last layer of the architecture used, the data is transformed into a linear vector, and in the classification layer, class information is generated by calculating the probabilistic values of each class (Singh et al., 2017).

3.2.1. Inception V3 Method

In general, the idea behind Inception modules is to increase model efficiency by segmenting the data differently from spatial and inter-channel correlation. Based on this idea, standard Inception modules first use 1x1 convolution blocks to evaluate inter-channel correlation. In the Inception modules, the input data can be mapped to different number of spaces with convolution windows of 3x3 or 5x5. This is illustrated in Fig. 2(a) in 3x3 dimensions.

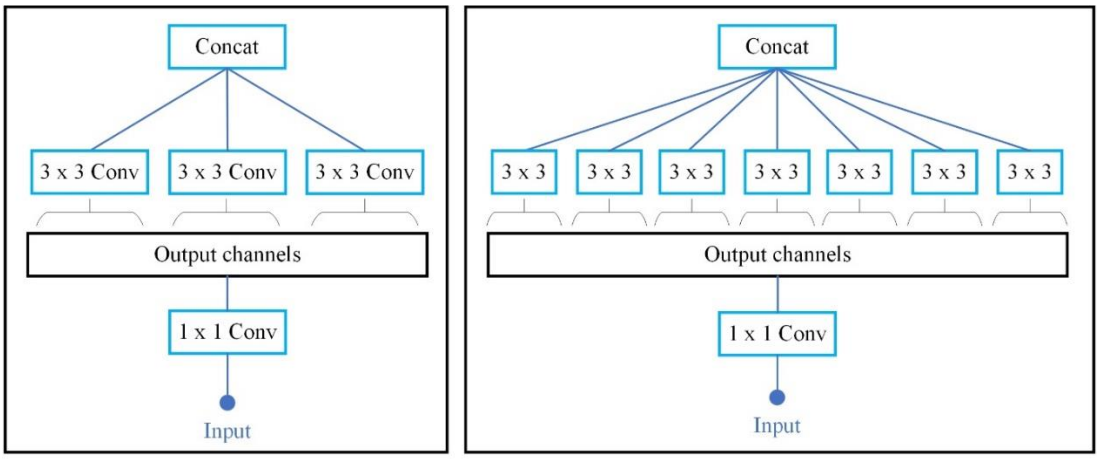


a) A simplified Inception module.

b) A strictly equivalent reformulation of the simplified Inception module.

Figure 2. a) standard form of Inception module, b) simplified version of the Inception module

The Inception V3 module has also been developed based on the Inception idea. This model focuses on the detailed decomposition of channel and spatial correlations. The Inception V3 model first consists of the canonical forms shown in Figure 2(a). Although the first modules of this architecture are similar to classical convolution layers, it can achieve highly discriminative representations with fewer parameters. In classical convolution layers, learning is performed with filters in a 3-dimensional space of width, height and channel size (Chollet 2017). Fig. 2(b) shows a simplified version of the Inception V3 module. Here, the 3x3 convolution dimension is used and mean pooling is not used.



a) A strict equivalent is shown of the simplified Inception module.

b) An "extreme" version of our Inception module, with one spatial convolution per output channel of the 1x1 convolution.

Figure 3. a) A depth wise separable convolution module, b) and extreme version of inception module

The structure representing the spatial convolution that will operate on the images in the output channels after the convolution process in 1x1 window sizes is shown in Fig. 3(a). The structure in the first 3 versions of the Inception V3 module is almost identical to the depth wise separable convolution introduced in 2014 and added to the tensorflow library in 2016 (Fime, Ashikuzzaman, and Aziz 2023). Apart from the first three versions of the Inception V3 module, the correlation between channels was mapped with a 1x1 window size filter and the correlation of each output channel was mapped separately. This is shown in detail in Figure 2.

In image processing applications, depth wise separable convolution process is called separable convolution. In this process, a single 1x1 point wise convolution is applied in the spatial convolution process performed on each channel of the input, and then 3x3 or 5x5 convolutions are applied to all outputs separately. These operations presented in Fig. 3(a) can be continued successively depending on the application.

Based on this strong hypothesis based on the inception module, a 1 x 1 convolution process is first applied and then the spatial correlations of each output channel are mapped. When these operations are applied in such a way as to produce a high degree of spatial collinearity, as shown in Fig. 3(b), an extreme inception structure is created. In this study, based on this structure, a new model is proposed, based on the Inception V3 module and basic inception stages.

3.2.2. Proposed Method

A new model is proposed based on Inception and Inception V3 modules. In the proposed model, as shown in Fig. 3(a), a separable structure is created by first applying [1×1] convolution process on the input image. In parallel with this, on the one hand, the input image was taken directly and avg, max and min pooling operations were carried out. At this stage, images were directly transferred to these filters and [1×1] convolution was not applied. The main reason for this operation is that whether a [1×1] convolution operation is applied or not, the data giving the largest, smallest and average value in the data will not change in the basic architecture. For this reason, [1×1] convolution operation was not performed. Thus, a positive contribution to the performance of the system in terms of time is provided. The proposed Inception module structure is presented in detail in Figure 4.

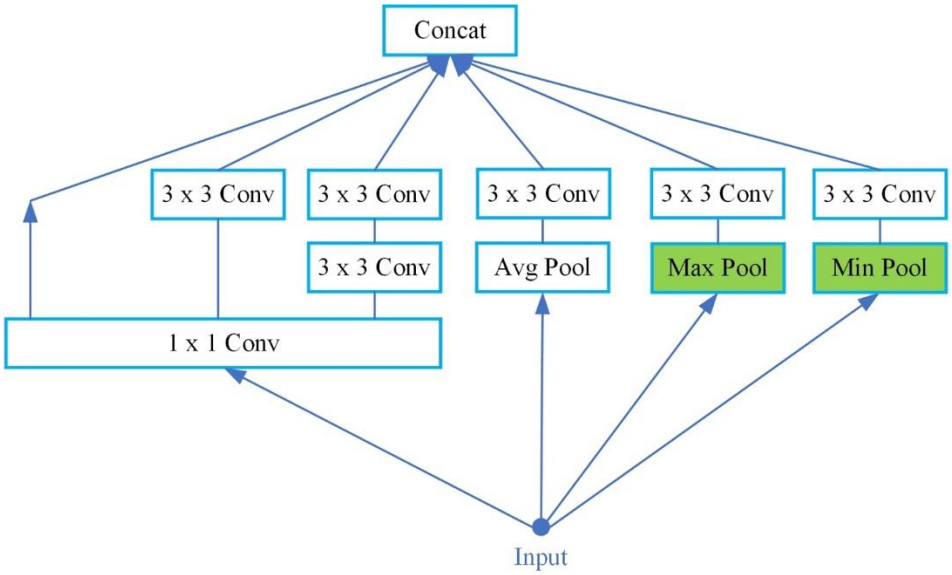


Figure 4. Inception SH inception module

The architecture created with the proposed module structure is shown in Figure 5. As can be seen in Fig. 5, the proposed architecture consists of a classification layer that includes batch normalization, dropout, fully connected and finally the softmax activation function. The batch normalization layer used in the proposed architecture normalizes the data. Therefore, there is no need to perform normalization operations again in the proposed module shown in Fig. 4. As a result, the performance of the proposed model is increased in terms of time and processing load. Again, in the dropout layer used in the proposed architecture, the system is prevented from storing the data. In the fully connected layer, all 2D data is converted into a one-dimensional vector to be presented to the classification layer. In the final layer of the system, the softmax classifier, which is widely used in the literature and whose accuracy has been proven in many studies, is used, presented in Eq.1.

$$softmax(c_j) = \frac{e^{C_j}}{\sum_{c=1}^{C} e^{z_c}} \; for \; j = 1, \ldots, C \qquad (1)$$

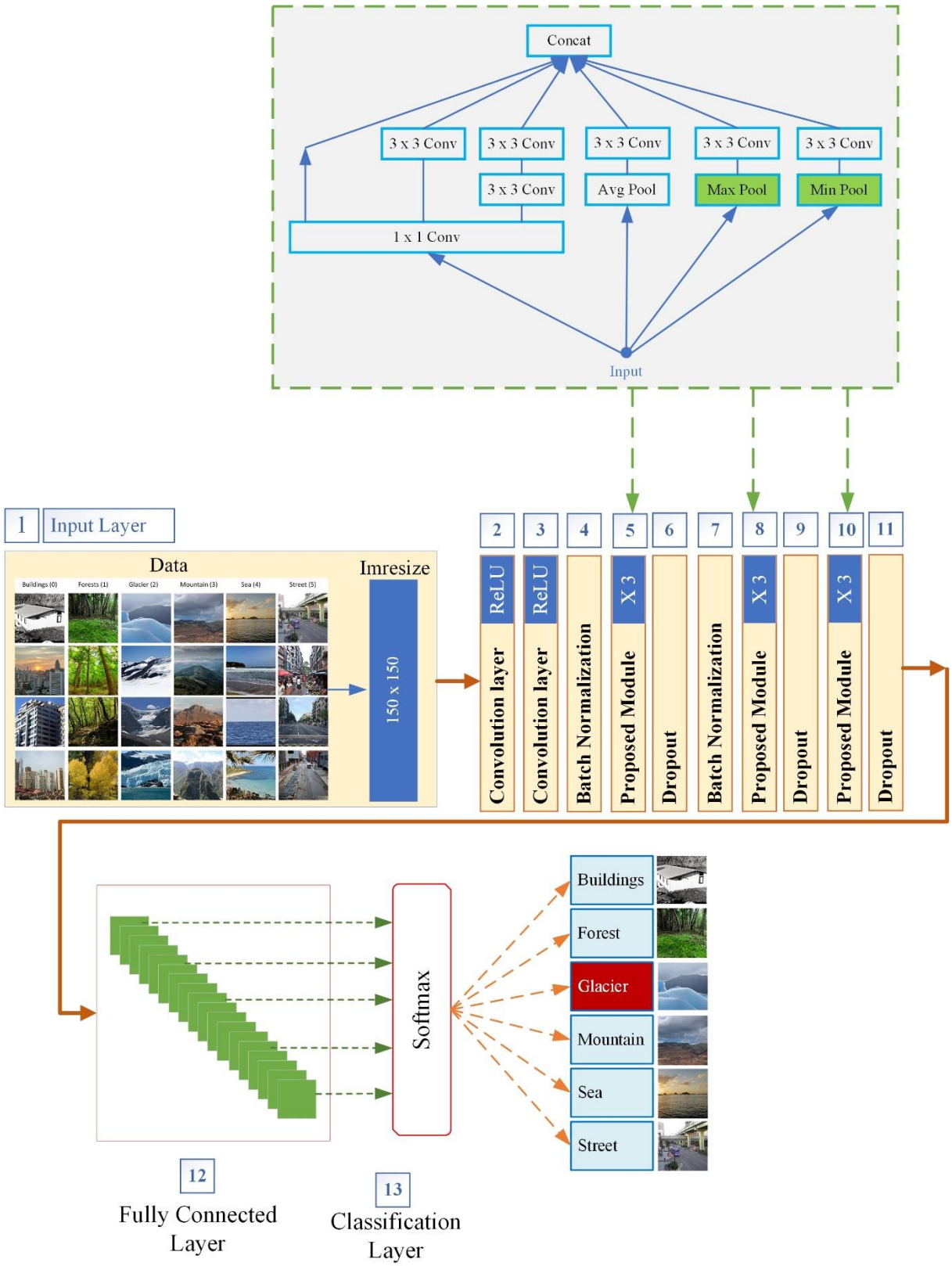


Figure 5. Inception SH architecture

Since it is known that the blocks used in the general architecture of the system will increase the processing load, the number of layers and the number of blocks were determined at an optimum level. In order to ensure that the proposed model is not specific to the dataset used in this study, an imresize operation is added in the first layer of the architecture. With the imresize operation, the images are resized to 150×150×3. However, since the dimensions of the images used in the study were already 150×150×3, the images in the dataset used in the study were not imresized.

Two convolutions with different stride values of [3×3] were performed to remove the noise on the image, especially due to environmental conditions. In order to avoid any change in the size values at the end of the convolution process, the padding value was set to the same in all convolution processes.

Table 2. Details of the layers of the Inception SH architecture

#	Layers	Filter count / Patch size / Stride	Input size	Output size
1	Input Layer	-	-	150x150x3
2	Convolution-1	32/3×3 / (2,2)	150x150x3	75x75x32
3	Convolution-2	64/3×3 / (1,1)	75x75x32	75x75x64
4	Batch Normalization	-	75x75x64	75x75x64
5	3 x Proposed Inception	4/3×3 / (1,1)	75x75x64	75x75x24
6	Dropout	0,2	75x75x24	75x75x24
7	Batch Normalization	-	75x75x24	75x75x24
8	3 x Proposed Inception	16/3×3 / (1,1)	75x75x24	75x75x96
9	Dropout	0,2	75x75x96	75x75x96
10	3 x Proposed Inception	64/3×3 / (1,1)	75x75x96	75x75x384
11	Dropout	0,2	75x75x384	75x75x384
12	Fully connected	256	75x75x384	75x75x256
13	Classifier	Softmax	75x75x256	75x75x6

All layers used in the proposed model, number of filters, patch sizes, stride values, input and output dimensions of each layer are presented in detail in Table 1. As can be seen in Table 1, 0.2 is used as the dropout value in the study. This value was tested between 0.1 and 0.8 in the dataset, and 0.2 was determined as the optimum value. The filter sizes used in the study were tested as [3×3] and [5×5]. As a result of the test, the [5×5] filter size trained the system faster than the [3×3] filter size, but the success rate decreased. Therefore, the [3×3] filter size was preferred in the study. The number of filters used in the proposed blocks was determined as a result of experimental studies. In the fully connected layer, a structure with 256 neurons was used to enhance the features. In the classification layer, which is the last stage, as many outputs are designed as the number of classes in the dataset used.

3.2.3. Evaluation Metrics

Accuracy, Recall, Precision, F1 score, and Categorical Cross-entrophy (CE_{Loss}) metrics are used in many classification applications in the literature to evaluate the performance of the developed deep learning models (Metlek & Çetiner 2023). Therefore, the same performance metrics are used in parallel with the literature. The contents of the performance metrics used in the study are presented in Eqs. 2- 6.

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (2)$$

$$Recall = \frac{TP}{TP+FN} \quad (3)$$

$$Precision = \frac{TP}{TP+FP} \quad (4)$$

$$F1\ score = 2x\frac{PrecisionxRecall}{Precision+Recall} \quad (5)$$

$$CE_{Loss} = -\frac{1}{N}\sum_{i=1}^{N}\sum_{c=1}^{C}(y_{ic}\ log\ (\widehat{y_{ic}})) \quad (6)$$

Eq. 6, y_{ic} is the classification result at the end of *i.* training for the c_{th} category, $\widehat{y_{ic}}$ is the probabilistic prediction result (Rusiecki 2019). Similarly, TP (True Positive), TN (True Negative), FN (False Negative), and FP (False Positive) are used in Eqs. 2-4.

4. Result and Discussion

The training results of the model proposed in the study and the model based on it are presented in detail in Table 3. As seen in Table 3, the accuracy, recall, precision and CE_{Loss} values of the proposed model are better than the underlying Inception V3. If these metrics are examined separately; The Accuracy value of the proposed model is approximately 0.07 higher than the Inception V3. When the recall values were examined, the proposed model showed a 0.14 higher performance than Inception V3. When the precision values are compared, the precision value of the proposed model showed 0.11 higher performance than Inception V3. In CE_{Loss} values, the proposed model gave approximately 0.06 lower loss value to Inception V3.

Table 3. Training performance results of Inception SH and Inception V3 models

	Accuracy	Recall	Precision	CE_{Loss}
Proposed	0.970	0.971	0.983	0.025
Inception V3	0.904	0.830	0.872	0.16

The results of the experimental study conducted with test data separated according to the cross-validation value of 5, unlike train data, are presented in detail in Table 4. As can be seen in Table 4, in the class-based performance results of the Inception V3 model, the best precision value was obtained in the Forest class, while the worst precision value was obtained in the Sea class. However, here the difference between the two results is not very high. As for the recall value, the recall metrics of all classes are closer to each other than the values in the precision metric. In the recall metric, the highest performance value was obtained from the Sea class, while the Forest class gave the lowest performance value. When the F1 score values are examined, it is seen that the F1 score metrics obtained from all classes are very close to each other. When the F1 score values of the Glacier and Forest classes were examined, it was determined that the results were closer to each other than the F1 score values of the other classes. As a result of the experimental procedures performed using the Inception V3 model, an average accuracy value of 0.882 was reached.

Table 4. Class-based test performance results of the Inception V3 model

Class	Accuracy	Precision	Recall	F1 score
Buildings	0.872	0.890	0.872	0.881
Forest	0.860	0.913	0.860	0.886
Glacier	0.889	0.887	0.889	0.888
Mountain	0.887	0.867	0.887	0.877
Sea	0.899	0.868	0.899	0.883
Street	0.886	0.874	0.886	0.880
Average	0.882	0.883	0.882	0.882
Avg. CE_{Loss}	0.098			

The performance results of the Inception SH model are also presented in detail in Table 5. As can be seen from here, precision, recall, and F1 score values are presented on a class basis, and accuracy and CE_{Loss} values are shared as the average of the classes. When the precision values of the proposed model are examined, it gives the same and higher performance values in the Forest and Glacier classes than the other classes. The lowest precision value was obtained from the mountain class. When the recall values are examined, the highest values were obtained from the buildings class and the lowest values were obtained from the mountain class. When the F1 score values were examined, the highest and same performance values were obtained from the buildings and street classes. The lowest F1 score value was obtained from the mountain class.

Table 5. Class-based test performance results of the proposed model

Class	Accuracy	Precision	Recall	F1 score
Buildings	0.977	0.957	0.977	0.967
Forest	0.947	0.961	0.947	0.954
Glacier	0.945	0.961	0.945	0.953
Mountain	0.942	0.953	0.942	0.948
Sea	0.966	0.955	0.966	0.961
Street	0.974	0.960	0.974	0.967
Average	0.958	0.957	0.974	0.967
Avg. CE_{Loss}	0.029			

In the study, the accuracy values of the proposed model were evaluated in two aspects: class-based and general. When the accuracy value was first examined on a class basis, the highest accuracy value was obtained from the buildings class. The lowest accuracy value was obtained from the mountain class. When the accuracy value was evaluated secondarily, the average accuracy value of all classes was measured as 0.958. The loss value of the proposed model was evaluated in general and was measured as 0.029.

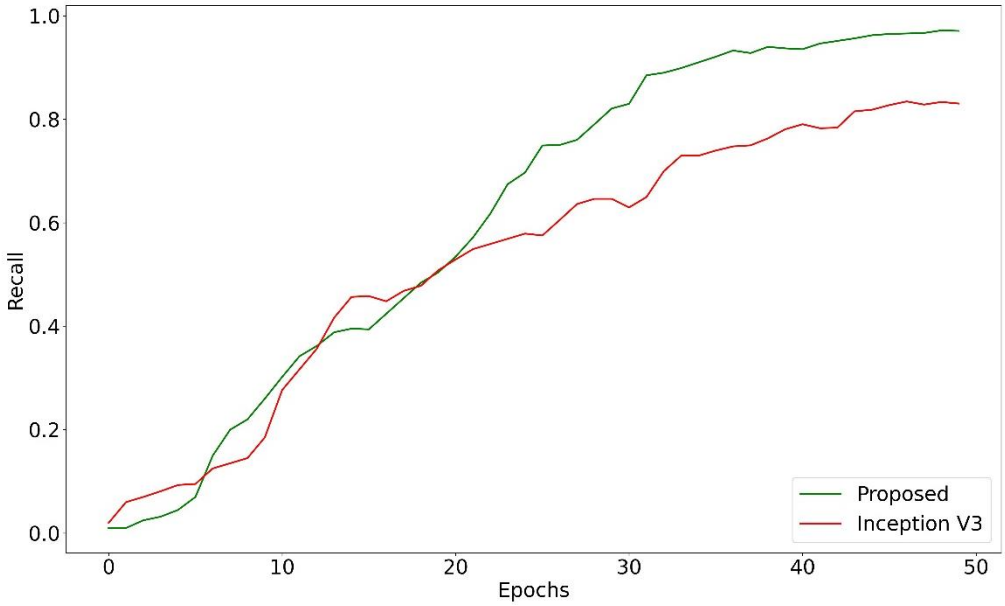


Figure 6. Train recall results of Inception SH and Inception V3 models

Among the training performance results presented in detail in Table 3, the performance of the recall value after 50 iterations is also presented visually in Figure 6. When the graph is examined in detail, despite the increase in the number of iterations, the superiority of the proposed model over the Inception V3 model continues.

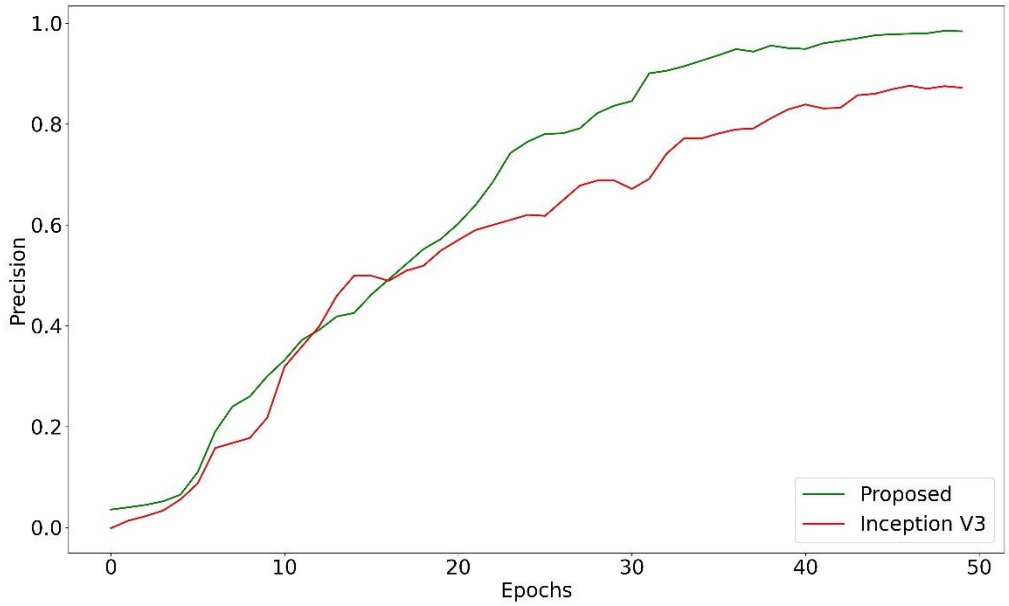


Figure 7. Train and validation precision results of Inception SH model

The graph of the precision value, which is presented similarly to the recall value in Table 3, is presented in Fig. 7. In this graph, similar to Fig. 6, despite the increase in the number of iterations, the superiority of the Inception SH model over the Inception V3 model continues. In the graph of the general accuracy value presented in Fig. 8, results similar to Fig. 6 and Fig. 7 stand out. In Figure 8, unlike these, it can be seen that the difference between the proposed model and the Inception V3 model is less.

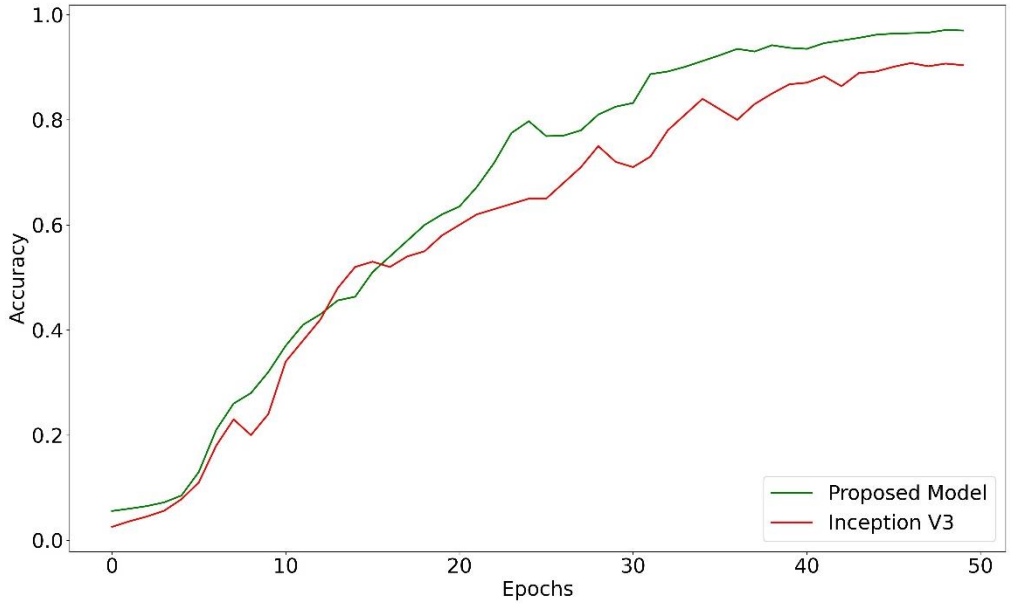


Figure 8. Train and validation accuracy results of Inception SH model

In Figure 9, the CE_{Loss} chart is presented, different from the accuracy, precision, and recall values. In this graph, it is seen that the loss value decreases as the number of iterations increases and the oscillation decreases and reaches a plateau, especially between 40 and 50 iterations. When these values are examined, it is seen that the proposed model provides a lower loss value than the Inception V3 model.

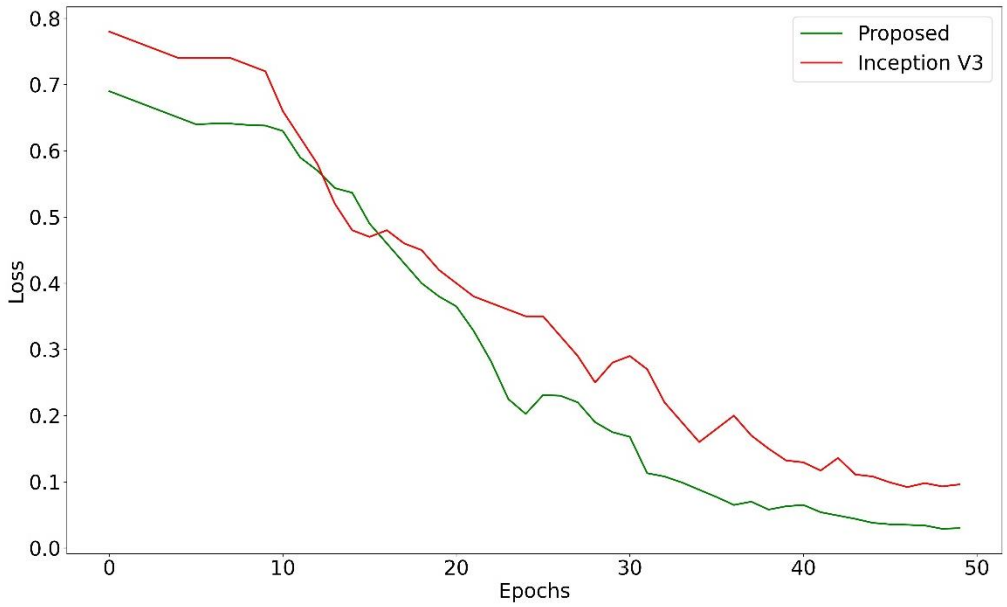


Figure 9. Train and validation CE_{Loss} results of Inception SH model

The confusion matrix, which is generally used in the literature, briefly summarizes the performance of a classification algorithm. This summary gives an idea of how accurate the predictions are and how close they are to the actual values. In the confusion matrix, rows are about real classes and columns are about predicted classes. The diagonal values in the confusion matrix represent correctly classified observations. It also shows that observations with off-diagonal values are misclassified. The class-based results obtained as a result of the testing process in the study are shared in detail in the confusion matrix presented in Figure 10.

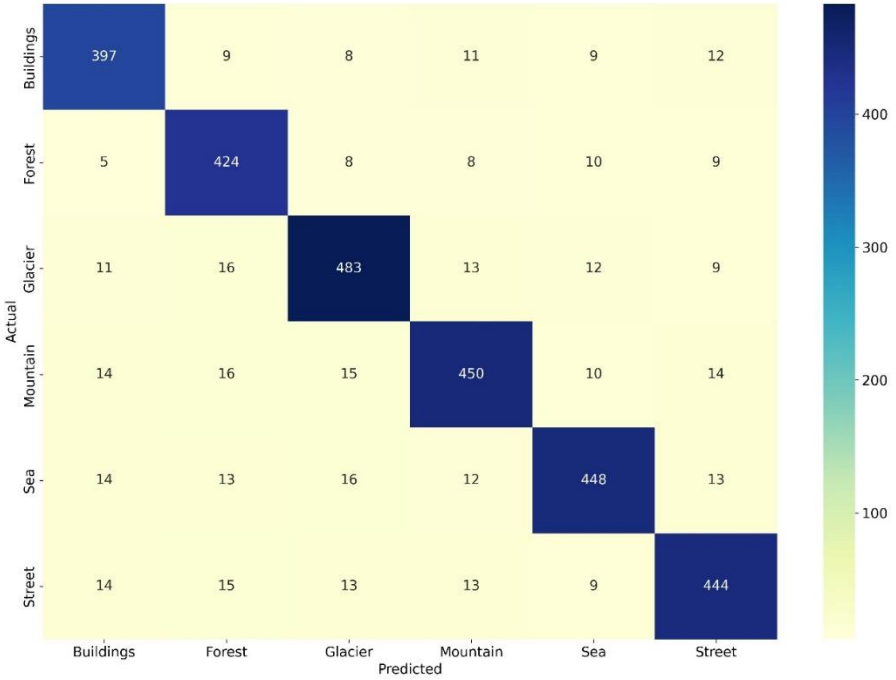


Figure 10. Confusion matrix of Inception V3

The class-based results obtained as a result of the testing processes are shared in the confusion matrix presented in Fig. 11. The values given in Figure 10 and Figure 11 summarize the values in Tables 3 and 4, respectively.

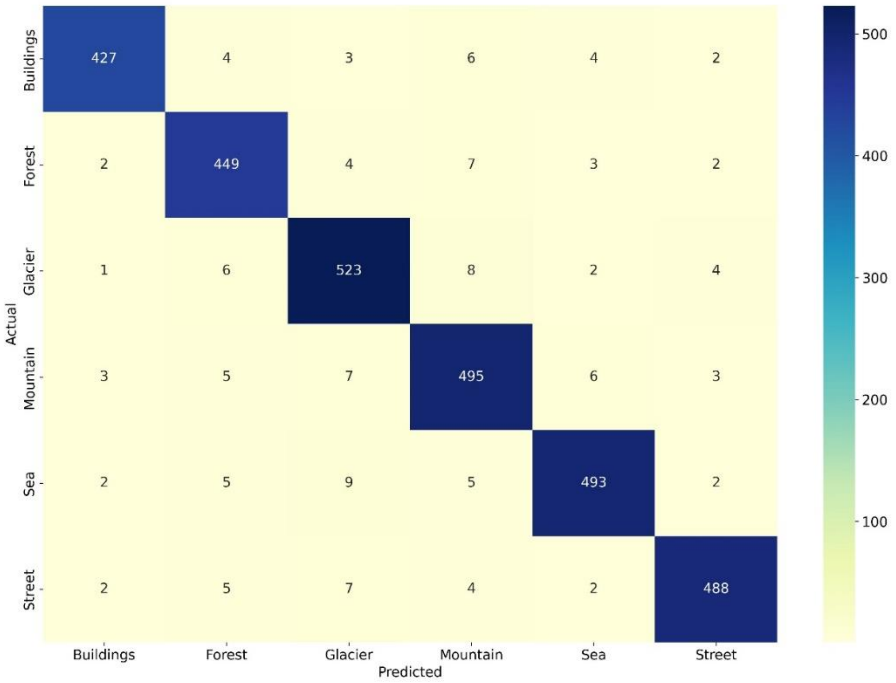


Figure 11. Confusion matrix of Inception SH model

Although existing studies in the literature regarding the dataset used in the study were investigated, it was determined that there were not many studies using the same dataset. Basic studies and performance values using the same dataset in the literature are briefly summarized in Table 6.

Table 6. State-of-the-art studies using of the same dataset

Authors [References]	Architecture	Accuracy	Recall	Precision	F1 Score
Sobti et al. (Sobti, Nayyar, and Nagrath 2021)	ResNet50	37	37	32	29
	VGG16	89	89	89	89
	VGG19	87	87	87	87
	Xception	90	90	90	90
	EnsemV3X	91	91	91	91
Guo et al. (Guo et al. 2021)	MobileNet	84	-	-	-
	MobileNet V2	86	-	-	-
	ShuffleNet (1.5)	85	-	-	-
	ShuffleNet (x2)	87	-	-	-
	MinorNet	88	-	-	-
Yahya et al. (Yahya et al. 2023)	ResNet50	83	-	-	-
	Xception	73	-	-	-
	DenseNet201	83	-	-	-
	NMAF	88	-	-	-
Chowdhury et al. (Chowdhury et al. 2022)	VGG16-Grid search	73	-	-	-
	VGG16-Genetic algorithm	78	-	-	-
	VGG-Bayesian optimization	84	-	-	-
	VGG16-Random search	83	-	-	-
	VGG16-Hyberband	82	-	-	-
	VGG16-Particle swarm optimization	81	-	-	-
Saran et al. (Saran, Saran, and Nar 2021)	Density Preserving Data Augmentation (DPDA)	90	-	-	-
	Flip Image (FI)	88	-	-	-
	Utilizing random erase	89	-	-	-
	DPDA+FI	88	-	-	-
	Gamma Correction (GC)	87	-	-	-
	Histogram Equalization Combined with Gamma Correction (HE+GC)	87	-	-	-
Thepade and Idhate (Thepade and Idhate 2022)	Random tree	65	-	-	-
	Random forest	78	-	-	-
	Simple logistic	81	-	-	-
	Logistic	82	-	-	-
	Naive Bayes	77	-	-	-
	Bayes net	80	-	-	-
	Multilayer perceptron	83	-	-	-
Ours	**Inception SH**	**95**	**95**	**95**	**95**

Using the same dataset as the one in the study, Sobti et al. used ResNet50, VGG16, VGG19, Xception, and EnsemV3X architectures in their study. Although the highest accuracy value was obtained from the EnsemV3X architecture, the lowest accuracy value was obtained from the ResNet50 architecture. EnsemV3X architecture, which gave high accuracy values in their studies, similarly showed higher performance than other architectures used in their studies in recall, precision, and F1-Score values.

Gua et al. also used MobileNet, MobileNet V2, ShuffleNet(1,5), ShuffleNet(x2), and MinorNet architectures in their study (Guo et al. 2021). They obtained the highest performance values from MinorNet. Yahya et al. performed classification using ResNet50, Xception, DenseNet201, and NMAF architectures (Yahya et al. 2023). While they achieved the highest result with the NMAF model they developed, they achieved the lowest result with the Xception architecture. Chowdhury et al classified the Intel image dataset with architectures named VGG16-Grid Search, VGG16-Genetic algorithm, VGG-Bayesian optimization, VGG16-Random search, VGG16-Hyberband, VGG16-Particle swarm optimization (Chowdhury et al. 2022). As a result of the classification, the highest performance result was obtained with the VGG16-Bayesian optimization method, while the lowest result was obtained with the VGG16-Grid search method. Saran et al carried out the classification process with methods called DPDA, FI, Utilizing random erase, DPDA+FI, GC, HE+GC (Saran et al. 2021). While the highest classification performance was achieved in the DPDA method, the lowest classification success was achieved in the GC and HE+GC models.

Thepade and Idhate performed classification with a random tree, random forest, simple logistic, logistic, naïve Bayes, Bayes net, and multilayer perceptron models (Thepade and Idhate 2022). In the performance results obtained as a result of the classification process, the lowest performance result was obtained with the random tree model, while the highest classification result was obtained with the multilayer perceptron model. When the studies in the literature using the same dataset are generally evaluated, it can be seen that their success performance is lower than the proposed model. In addition, in many studies, the authors did not share the values of the Recall, Precision, and F1 Score performance metrics in their studies.

Table 7. Parameters numbers and sizes of the models

Model	#Parameters (M)	Size (MB)
Inception (Zhu et al. 2023)	10.32	39.42
Inception V3 (Cao et al. 2021; Pan et al. 2023)	21.8	83.89
Inception SH	5,56	24.72

The number of parameters according to the Inception and Inception V3 models inspired by the study is also presented in detail in Table 7. As can be seen from here, the number of parameters is less than the other two models. It is obvious that having fewer parameters will reduce the computational cost. As a result, it's qualification more applicable in embedded systems than the other two inspired models due to its lower processing cost.

5. Conclusion

In this study, a deep learning model based on Inception V3 has been developed to enable UAVs to make instant decisions about the environment by autonomously making instant decisions from images independently of the operators using the UAVs. The developed model was trained and tested on an up-to-date dataset named Intel Image Dataset. The dataset contains real-life images of Buildings, Forest, Glacier, Mountain, Sea and Street classes. Separable structure is created by applying [1×1] convolution process in the inception module structure used in the proposed model. In parallel with this, avg, max and min pooling operations were performed by taking the input image directly on one side. In this parallel process, images are directly transferred to these filters and [1×1] convolution process is not applied. Thus, a positive contribution has been made to the performance of the proposed architecture in terms of time. The general structure of this module is presented in detail in Fig. 4. This block structure and the proposed architecture are presented as a whole in Fig. 5. Although the proposed architecture consists of 13 steps in total, in the first step, the size of the images is reduced to 150x150 by performing the imresize operation especially in the input layer. The number of layers and blocks used in the overall architecture of the system was determined by experimental studies and kept at an optimum level. Thus, the proposed model is designed to work on embedded systems.

Accuracy, recall, precision, F1 score and CE_{Loss} metrics, which are preferred in the literature, were used to evaluate the performance of the study. Although the proposed system is based on the Inception V3 model, the difference in performance is detailed in Tables 3-5. In addition, a detailed comparison with studies in the literature using the same dataset is presented in Table 6. The proposed model provides a superior performance result compared to the studies using the same dataset. The proposed model achieves 95% performance on average, while the closest work in the literature achieves 91% (Sobti et al. 2021). In other words, the proposed model provides approximately 4% better results compared to the accuracy, recall, precision, and F1 score metrics commonly used in the literature.

As a result, the proposed architecture is an improvement in terms of performance compared to the Inception V3 model. Since the proposed model is lighter than many models in the literature, it is obvious that it can be used on different platforms. It is aimed to adapt the study to different subjects by making changes in the filters in the block in the future. When all the results are evaluated, the results obtained are competitive with other studies in the literature. Considering the popularity of autonomous UAVs, it is predicted that the Inception SH model, developed as a lightweight model, can also be used in IoT devices.

Conflict of Interest

No conflict of interest was declared by the authors.

References

Akbay, Tuncer. 2022. Modeling Education Studies Indexed in Web of Science Using Natural Language Processing. Instructional Technology and Lifelong Learning 3(2):129–43.

Amarasingam, Narmilan, Arachchige Surantha Ashan Salgadoe, Kevin Powell, Luis Felipe Gonzalez, and Sijesh Natarajan. 2022.

A Review of UAV Platforms, Sensors, and Applications for Monitoring of Sugarcane Crops. Remote Sensing Applications: Society and Environment 26:100712.

Cao, Jianfang, Minmin Yan, Yiming Jia, Xiaodong Tian, and Zibang Zhang. 2021. Application of a Modified Inception-v3 Model in the Dynasty-Based Classification of Ancient Murals. EURASIP Journal on Advances in Signal Processing 2021:1–25.

Çetiner, Halit, and Sedat Metlek. 2023. DenseUNet+: A Novel Hybrid Segmentation Approach Based on Multi-Modality Images for Brain Tumor Segmentation. Journal of King Saud University - Computer and Information Sciences 35(8):101663. doi: https://doi.org/10.1016/j.jksuci.2023.101663.

Chollet, François. 2017. Xception: Deep Learning with Depthwise Separable Convolutions. Pp. 1251–58 in Proceedings of the IEEE conference on computer vision and pattern recognition.

Chowdhury, Anjir Ahmed, Argho Das, Khadija Kubra Shahjalal Hoque, and Debajyoti Karmaker. 2022. A Comparative Study of Hyperparameter Optimization Techniques for Deep Learning BT - Proceedings of International Joint Conference on Advances in Computational Intelligence. Pp. 509–21 in, edited by M. S. Uddin, P. K. Jamwal, and J. C. Bansal. Singapore: Springer Nature Singapore.

Fime, Awal Ahmed, Md Ashikuzzaman, and Abdul Aziz. 2023. Audio Signal Based Danger Detection Using Signal Processing and Deep Learning. Expert Systems with Applications 121646.

Finnegan, Philip. 2017. World Civil Unmanned Aerial Systems Market Profile and Forecast 2017. Teal Group 1–13. Retrieved (https://tealgroup.com/images/TGCTOC/WCUAS2017TOC_EO.pdf).

Gómez-Chova, L., D. Tuia, G. Moser, and G. Camps-Valls. 2015. Multimodal Classification of Remote Sensing Images: A Review and Future Directions. Proceedings of the IEEE 103(9):1560–84. doi: 10.1109/JPROC.2015.2449668.

Grand View, Research. 2023. Commercial UAV Market Size, Share & Trends Analysis Report By Product (Fixed Wing, Rotary Blade, Nano, Hybrid), By Application (Agriculture, Energy, Government, Media & Entertainment, Construction), By Region, And Segment Forecasts, 2023 - 2030. Grand View Research 171. Retrieved (https://www.grandviewresearch.com/industry-analysis/commercial-uav-market).

Guo, S., Y. Ni, K. Xing, Y. Liu, and W. Ni. 2021. MinorNet: A Lightweight Neural Network for Battlefield Scene Classification. Pp. 17–20 in 2021 14th International Symposium on Computational Intelligence and Design (ISCID).

Hu, Fan, Gui-Song Xia, Jingwen Hu, and Liangpei Zhang. 2015. Transferring Deep Convolutional Neural Networks for the Scene Classification of High-Resolution Remote Sensing Imagery. Remote Sensing 7(11):14680–707.

Huang, Rachel, Jonathan Pedoeem, and Cuixian Chen. 2018. YOLO-LITE: A Real-Time Object Detection Algorithm Optimized for Non-GPU Computers. Pp. 2503–10 in 2018 IEEE international conference on big data (big data). IEEE.

Li, Miao, Shuying Zang, Bing Zhang, Shanshan Li, and Changshan Wu. 2014. A Review of Remote Sensing Image Classification Techniques: The Role of Spatio-Contextual Information. European Journal of Remote Sensing 47(1):389–411. doi: 10.5721/EuJRS20144723.

Matese, Alessandro, Piero Toscano, Salvatore F. Di Gennaro, Lorenzo Genesio, Francesco P. Vaccari, Jacopo Primicerio, Claudio Belli, Alessandro Zaldei, Roberto Bianconi, and Beniamino Gioli. 2015. Intercomparison of UAV, Aircraft and Satellite Remote Sensing Platforms for Precision Viticulture. Remote Sensing 7(3):2971–90.

Maulik, U., and D. Chakraborty. 2017. Remote Sensing Image Classification: A Survey of Support-Vector-Machine-Based Advanced Techniques. IEEE Geoscience and Remote Sensing Magazine 5(1):33–52. doi: 10.1109/MGRS.2016.2641240.

Menouar, H., I. Guvenc, K. Akkaya, A. S. Uluagac, A. Kadri, and A. Tuncer. 2017. UAV-Enabled Intelligent Transportation Systems for the Smart City: Applications and Challenges. IEEE Communications Magazine 55(3):22–28. doi: 10.1109/MCOM.2017.1600238CM.

Metlek, S, and H. Çetiner. 2023. ResUNet+: A New Convolutional and Attention Block-Based Approach for Brain Tumor Segmentation. IEEE Access 11:69884–902. doi: 10.1109/ACCESS.2023.3294179.

Metlek, Sedat, and Halit Çetiner. 2023. Classification of Poisonous and Edible Mushrooms with Optimized Classification Algorithms. Pp. 408–15 in International Conference on Applied Engineering and Natural Sciences. Vol. 1.

Moranduzzo, T., F. Melgani, M. L. Mekhalfi, Y. Bazi, and N. Alajlan. 2015. Multiclass Coarse Analysis for UAV Imagery. IEEE Transactions on Geoscience and Remote Sensing 53(12):6394–6406. doi: 10.1109/TGRS.2015.2438400.

Noble, William S. 2006. What Is a Support Vector Machine? Nature Biotechnology 24(12):1565–67. doi: 10.1038/nbt1206-1565.

Nogueira, Keiller, Otávio A. B. Penatti, and Jefersson A. dos Santos. 2017. Towards Better Exploiting Convolutional Neural Networks for Remote Sensing Scene Classification. Pattern Recognition 61:539–56. doi: https://doi.org/10.1016/j.patcog.2016.07.001.

Pan, Yuhang, Junru Liu, Yuting Cai, Xuemei Yang, Zhucheng Zhang, Hong Long, Ketong Zhao, Xia Yu, Cui Zeng, Jueni Duan, Ping Xiao, Jingbo Li, Feiyue Cai, Xiaoyun Yang, and Zhen Tan. 2023. Fundus Image Classification Using Inception V3 and ResNet-50 for the Early Diagnostics of Fundus Diseases. Frontiers in Physiology 14.

Penatti, O. A. B., K. Nogueira, and J. A. dos Santos. 2015. Do Deep Features Generalize from Everyday Objects to Remote Sensing and Aerial Scenes Domains? Pp. 44–51 in 2015 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW).

Quinlan, J. Ross. 1986. Induction of Decision Trees. Machine Learning 1:81–106.

Rusiecki, A. 2019. Trimmed Categorical Cross-Entropy for Deep Learning with Label Noise. Electronics Letters 55(6):319–20. doi: https://doi.org/10.1049/el.2018.7980.

Saran, Nurdan Ayse, Murat Saran, and Fatih Nar. 2021. Distribution-Preserving Data Augmentation. PeerJ Computer Science 7:e571.

Şenel, Bilge, and Fatih Ahmet Şenel. 2022. Novel Neural Network Optimization Approach for Modeling Scattering and Noise Parameters of Microwave Transistor. International Journal of Numerical Modelling: Electronic Networks, Devices and Fields 35(1):e2930.

Shabbir, Amsa, Nouman Ali, Jameel Ahmed, Bushra Zafar, Aqsa Rasheed, Muhammad Sajid, Afzal Ahmed, and Saadat Hanif Dar. 2021. Satellite and Scene Image Classification Based on Transfer Learning and Fine Tuning of ResNet50. edited by M. Maqsood. Mathematical Problems in Engineering 2021:5843816. doi: 10.1155/2021/5843816.

Shahi, Tej Bahadur, Cheng-Yuan Xu, Arjun Neupane, and William Guo. 2023. Recent Advances in Crop Disease Detection Using UAV and Deep Learning Techniques. Remote Sensing 15(9):2450.

Singh, Vineeta, Deeptha Girish, and Anca L. Ralescu. 2017. Image Understanding-a Brief Review of Scene Classification and Recognition. MAICS 2017:85–91.

Sobti, Priyal, Anand Nayyar, and Preeti Nagrath. 2021. EnsemV3X: A Novel Ensembled Deep Learning Architecture for Multi-Label Scene Classification. PeerJ Computer Science 7:e557.

Thepade, Sudeep D., and Mrunal E. Idhate. 2022. Machine Learning-Based Scene Classification Using Thepade's SBTC, LBP, and GLCM BT - Futuristic Trends in Networks and Computing Technologies. Pp. 603–12 in, edited by P. K. Singh, S. T. Wierzchoń, J. K. Chhabra, and S. Tanwar. Singapore: Springer Nature Singapore.

Tokmak, Mahmut. 2022. Uzun-Kısa Süreli Bellek Ağı Kullanarak Hisse Senedi Fiyatı Tahmini. Mehmet Akif Ersoy Üniversitesi Uygulamalı Bilimler Dergisi 6(2):309–22.

Tuia, D., M. Volpi, L. Copa, M. Kanevski, and J. Munoz-Mari. 2011. A Survey of Active Learning Algorithms for Supervised Remote Sensing Image Classification. IEEE Journal of Selected Topics in Signal Processing 5(3):606–17. doi: 10.1109/JSTSP.2011.2139193.

Wu, X., R. Liu, H. Yang, and Z. Chen. 2020. An Xception Based Convolutional Neural Network for Scene Image Classification with Transfer Learning. Pp. 262–67 in 2020 2nd International Conference on Information Technology and Computer Application (ITCA).

Xia, Gui-Song, Jingwen Hu, Fan Hu, Baoguang Shi, Xiang Bai, Yanfei Zhong, Liangpei Zhang, and Xiaoqiang Lu. 2017. AID: A Benchmark Data Set for Performance Evaluation of Aerial Scene Classification. IEEE Transactions on Geoscience and Remote Sensing 55(7):3965–81. doi: 10.1109/TGRS.2017.2685945.

Yahya, Ali A., Kui Liu, Ammar Hawbani, Yibin Wang, and Ali N. Hadi. 2023. A Novel Image Classification Method Based on Residual Network, Inception, and Proposed Activation Function. Sensors 23(6).

Yuan, C., Z. Liu, and Y. Zhang. 2015. UAV-Based Forest Fire Detection and Tracking Using Image Processing Techniques. Pp. 639–43 in 2015 International Conference on Unmanned Aircraft Systems (ICUAS).

Zeggada, A., and F. Melgani. 2017. Multilabeling UAV Images with Autoencoder Networks. Pp. 1–4 in 2017 Joint Urban Remote Sensing Event (JURSE).

Zhang, L., L. Zhang, and B. Du. 2016. Deep Learning for Remote Sensing Data: A Technical Tutorial on the State of the Art. IEEE Geoscience and Remote Sensing Magazine 4(2):22–40. doi: 10.1109/MGRS.2016.2540798.

Zhu, Xianyu, Jinjiang Li, Ruchang Jia, Bin Liu, Zhuohan Yao, Aihong Yuan, Yinqiu Huo, and Zhang Haixi. 2023. LAD-Net: A Novel Light Weight Model for Early Apple Leaf Pests and Diseases Classification. IEEE/ACM Transactions on Computational Biology And Bioinformatics 20(2):1156–69.

Zou, Q., L. Ni, T. Zhang, and Q. Wang. 2015. Deep Learning Based Feature Selection for Remote Sensing Scene Classification. IEEE Geoscience and Remote Sensing Letters 12(11):2321–25. doi: 10.1109/LGRS.2015.2475299.