remote sensing

Article

Remote Sensing Image Scene Classification with Self-Supervised Learning Based on Partially Unlabeled Datasets

Xiliang Chen , Guobin Zhu * and Mingqing Liu

School of Remote Sensing and Information Engineering, Wuhan University, Wuhan 430079, China
* Correspondence: gbzhu@whu.edu.cn; Tel.: +86-027-6877-8916

Abstract: In recent years, supervised learning, represented by deep learning, has shown good performance in remote sensing image scene classification with its powerful feature learning ability. However, this method requires large-scale and high-quality handcrafted labeled datasets, which leads to a high cost of obtaining annotated samples. Self-supervised learning can alleviate this problem by using unlabeled data to learn the image's feature representation and then migrate to the downstream task. In this study, we use an encoder–decoder structure to construct a self-supervised learning architecture. In the encoding stage, the image mask is used to discard some of the image patches randomly, and the image's feature representation can be learned from the remaining image patches. In the decoding stage, the lightweight decoder is used to recover the pixels of the original image patches according to the features learned in the encoding stage. We constructed a large-scale unlabeled training set using several public scene classification datasets and Gaofen-2 satellite data to train the self-supervised learning model. In the downstream task, we use the encoder structure with the masked image patches that have been removed as the backbone network of the scene classification task. Then, we fine-tune the pre-trained weights of self-supervised learning in the encoding stage on two open datasets with complex scene categories. The datasets include NWPU-RESISC45 and AID. Compared with other mainstream supervised learning methods and self-supervised learning methods, our proposed method has better performance than the most state-of-the-art methods in the task of remote sensing image scene classification.

Keywords: self-supervised learning; vision transformer; random mask; remote sensing image scene classification; unlabeled datasets

check for updates

Citation: Chen, X.; Zhu, G.; Liu, M. Remote Sensing Image Scene Classification with Self-Supervised Learning Based on Partially Unlabeled Datasets. *Remote Sens.* **2022**, *14*, 5838. https://doi.org/10.3390/rs14225838

Academic Editors: Jia Wan, Zhitong Xiong and Jiangbin Zheng

Received: 6 October 2022
Accepted: 15 November 2022
Published: 18 November 2022

1. Introduction

With the rapid development and progress of remote sensing technology, more and more remote sensing images with high-resolution can be obtained easily. In addition to rich spatial and texture information, these remote sensing images also contain a lot of potential semantic information. The development trend promotes the application of remote sensing data as useful information in many fields, such as land resource management and urban planning [1,2]. Because of the limitation of low spatial resolution, early remote sensing image classification methods were mainly based on pixels and targets [3]. Because the pixels or targets only contain low-level local feature information, it is impossible to obtain semantic information (such as industrial areas, commercial areas, schools, etc.) on the scene level. With the increasing spatial resolution of remote sensing images, scene-level image classification combined with the context information in a larger interpretation unit has become a hot research issue [4]. The target of this research is to meet the requirements of higher-level remote sensing image interpretation and bridge the "semantic gap" between local features in low-level and global scene semantic features in high-level. Each remote sensing image is composed of different ground objects. Through high spatial resolution remote sensing image scene classification technology, we can accurately assign a unique scene category to each remote sensing image based on global semantic information [5]. At present, handcrafted feature extraction, feature coding, and deep learning are three main methods of remote sensing image scene classification. Commonly used handcrafted features include color histogram [6], gray level co-occurrence matrix (GLCM) [7], Gabor feature [8], local binary pattern (LBP) [9], scale-invariant feature transform (SIFT) [10] feature, histogram of oriented gradient (HOG) [11] feature and GIST [12] feature. Although these features are stable and easy to calculate, when the spatial distribution of object images is very complex, the discriminability of these features may be greatly reduced. For example, low-density residential areas and high-density residential areas are easily confused, and churches and palaces have many similar structures. Therefore, the effect of scene classification is very poor. Compared with the low-level features extracted directly, the feature coding methods which belong to the middle-level features have a stronger learning ability and can express more abundant image information. Commonly used feature coding methods include bag of visual words (BOVW) [13], vector locally aggregated descriptors (VLAD) [14] and fisher vector (FV) [15]. The feature coding methods have considered more comprehensive information, which have improved the semantic understanding. Because the essence of the feature encoding method is the further integration of the low-level features, which has a high dependence on the artificially designed features. As a result, the "semantic gap" between middle-level features and high-level semantic features of the image scene has not yet been eliminated.

In recent years, benefit from the development of artificial intelligence technology, deep learning methods with powerful feature learning ability and multi-level feature expression ability have been successfully applied in many fields which include remote sensing scene classification. Through deep learning methods, we can extract high-level features of images without manual intervention, and achieve “end-to-end” learning in the task of image target recognition and classification. Among many deep learning methods, convolutional neural network (CNN) is the most successful method applied in remote sensing image processing. Since then, the accuracy of remote sensing image scene classification has been greatly improved [1]. Early scene classification methods based on CNN usually need to completely train a new CNN model from scratch. Nevertheless, the CNN model depends on data-driven. It is known to all that deep CNN networks usually contain millions of trainable parameters and need to collect a large number of annotated training datasets [16]. However, the accurate annotation of remote sensing image scenes is tedious work, which requires rich experience and professional geographical knowledge. In the currently published remote sensing image scene classification datasets, although there exist large datasets containing millions of samples, the categories of scenes included are still very limited. As a representative example, MillionAID [17] is so far the largest remote sensing image dataset with a scale of millions of images similar to ImageNet-1K. However, the dataset only contains 51 scene categories, and compared with the 1000 scene categories of the ImageNet-1K dataset, there is a large gap between them. Owing to the scarcity of training datasets, methods based on CNN still face many challenges in remote sensing image interpretation tasks in a large area and complex scenes (such as global mapping [18]). To alleviate the dependence on labeled samples of remote sensing images, one of the most common methods is to use pre-trained convolutional neural networks on large-scale labeled natural image samples, such as the ImageNet dataset [19] with over 10 million labeled natural image samples, and then fine-tune the network in the way of transfer learning to complete remote sensing scene classification. Commonly used pre-trained networks include AlexNet [20], VGGNet [21], GoogleNet [22], ResNet [23], and CaffeNet [24]. In the case of insufficient samples, although the pre-trained model on ImageNet can be well generalized to the tasks of remote sensing image scene classification tasks and effectively alleviate the overfitting problem, there are still some deficiencies in the following aspects:

(1) The image generation mechanism of remote sensing images is different from natural images. Natural images usually only have three bands of RGB. In addition to visible light bands, remote sensing images may also have infrared bands. If the pre-trained network on

natural images is directly used for fine-tuning, spectral features outside the visible light range can not be fully exploited, which will lead to the network parameters not being optimized according to the unique features of remote sensing images.

(2) The structure of remote sensing images is more complex than the natural images. For remote sensing scene classification, we usually obtain the local image patches from large satellite images with several resolutions, and they are composed of highly complex geometric structures. Ground-truth objects vary widely in terms of perspective, scale, and shape, and at the same time, they have inter-class similarities and intra-class differences.

(3) Pre-training the network on the ImageNet dataset has a high time cost. When classifying a remote sensing scene, we usually use a model that has been trained on the ImageNet dataset, ignoring the time and computational cost of pre-training, which gives us the impression that the use of a pre-trained model can reduce the time cost.

Self-supervised learning is a method that was first successfully applied in the field of computer vision. It is the most promising way to solve the above problems. Self-supervised learning is a new machine learning paradigm that has become popular in recent years [25]. A significant difference between self-supervised learning and supervised learning is that self-supervised learning can mine its own supervision information through a large number of unlabeled data sets with rich self-supervised learning signals that were artificially designed. The self-supervised learning signals can be used to construct pseudo-labels, then replace the traditional manual labeled data to drive the model for feature learning. Self-supervised learning techniques have made great success in natural image analysis tasks, and even in some subdivision tasks (such as image object detection, medical image segmentation, and target tracking), the overall accuracy has approached or even exceeded supervised learning methods [26], but there is little research about remote sensing. In our paper, we propose a self-supervised learning architecture based on an encoder–decoder structure. Firstly, we divide the image uniformly into local patches and then flatten them into a sequence of patches. Then, in the encoding stage, some of the image patches are discarded by the image mask randomly, the image's feature representation can be learned from the remained image patches, and finally combined with the mask token of the image to form a joint encoding. In the decoding stage, a lightweight decoder is used to recover the original image according to the features learned in the encoding stage. The above process is performed on an unlabeled remote sensing dataset. Using remote sensing scene classification as a downstream task, we fine-tune the pre-trained weights in the encoding stage of self-supervised learning. Our contributions include the following aspects:

(1) To solve the problem of dependence on a large number of labeled remote sensing images in the supervised learning paradigm in scene classification, we introduce a self-supervised learning framework with an encoder–decoder structure, the method can learn its feature representation from unlabeled images.

(2) We introduce a strategy based on image patch recovery to train the self-supervised task. We randomly discard some of the image patches by using the image mask and learn the image features representation from the remained image patches and then combine them with the mask token of the image to form a joint encoding. Then, the image patches discarded in the encoder can be reconstructed. In addition, we also take into account the position information of the remained image patches itself, the position information of the remained image patches and the masked image remained relative to the original image remained sequence. Hence, two position embedding operations were performed, respectively.

(3) We constructed an unlabeled dataset for the training of the self-supervised learning task by using the public artificially produced scene classification training dataset and the Gaofen-2 satellite data without human intervention. The comparative experiments have been conducted on the two datasets, respectively, which proved that the unlabeled data produced without manual intervention can be used for self-supervised training, which is helpful to improve over accuracy of scene classification task.

The rest of the paper is organized as follows: the related work is briefly introduced in Section 2, which included transfer learning, few-shot learning and self-supervised learning. In Section 3, we introduce the proposed method. Section 4 presents the experiment results, and we draw a conclusion in Section 5.

2. Related Work

A. Transfer Learning

Transfer learning is a method of applying knowledge acquired in one domain to solve a new problem in another but related field. The purpose is to transfer existing knowledge to a specific domain with a small amount of annotated data. Because there are very few samples in new tasks, transfer learning can effectively use the existing data to solve the learning problems in new tasks. There are three ways to use the pre-trained CNN model in remote sensing image scene classification: full training, fine-tuning, and feature extraction. When fine-tuning the pre-trained CNN model through newly labeled training samples in the target field, the network has been established based on the optimized initialization parameters, and the training parameters can converge faster. Therefore, the fine-tuning method has strong adaptability and good performance in the task of remote sensing scene classification [27]. However, the method of fine-tuning is highly dependent on the existing pre-trained CNN network structure, which is not flexible enough. When the knowledge of the deep network is transferred to the small-scale training dataset, it still faces the problem of gradient disappearance [16]. The convolution layer or full connection layer of the pre-trained model can play a role of feature extractor. Combined with other machine learning methods, the extracted deep features can be further optimized. The features represent the global information of the image can be extracted through the fully connected (FC) layer, which contain rich high-level semantic information that can efficiently distinguish different scenes [28]. Chaib et al. [29] integrated different full connection layer features in CNN to strengthen the expression ability of image features. The approach of transfer learning has a wide range of applications in remote sensing scene classification, such as references [1,30–32].

B. Few-Shot Learning

The few-shot learning method has been a research hotspot in recent years, which can learn from the prior knowledge with few annotated datasets [33,34]. Currently, few-shot learning methods are mainly implemented based on meta-learning. Meta-learning is suitable for learning with few-samples and multi-task, which can solve the problems of rapid learning and rapid adaptation when new tasks lack training samples [35]. Meta learning requires the support of multiple different but related tasks. Each task has its own training set and test set [36]. To face the challenges of fast learning of new tasks with few samples, multiple tasks similar to the new tasks are necessary to be constructed, and they will be used as training sets to participate in meta-training [37,38]. The successful application of few-shot learning on natural image classification has attracted a large number of researchers in the field of remote sensing, who try to apply few-shot learning to remote sensing scene classification. Zhai et al. [39] proposed a model based on meta-learning method with gradient descent and has the particularity that knowledge learned from one data set can be easily and rapidly adapted to a new data set. Li et al. [40] proposed a model called discriminative learning of adaptive match network (DLA-MatchNet) which can automatically discover discriminative regions. The method can leverage an episode-based strategy to train the model. Once trained, the model can predict the category of query image without further fine-tuning. Li et al. [41] propose a method called RS-MetaNet, the method can raise the level of learning from the sample to the task by organizing training in a meta way, and it can learn to learn a metric space that can well classify remote sensing scenes from a series of tasks. Li et al. [42] introduced an end-to-end framework called self-supervised contrastive learning-based metric learning network (SCL-MLNet) for few-shot remote sensing (RS) scene classification. Although the approach of few-shot learning for remote sensing scene image classification can overcome the dependence on a large amount of training datasets, at present, the classification accuracy of this method is not high, and there is still a lot of room for improvement.

C. Self-supervised Learning

Self-supervised learning is a new machine learning paradigm emerging in recent years [25], which has attracted the attention of a large number of researchers. Unlike supervised learning, it can learn latent feature representation of images from massive unlabeled data through the agent task of self-supervised learning, which is artificially designed. The way of using unlabeled images for representation learning has become particularly attractive in the field of remote sensing because it is very difficult to obtain large-scale labeled samples. At the same time, the method can overcome the impact of differences between natural images and remote sensing images in the previous transfer learning process. Self-supervised learning techniques have been successfully applied in natural image processing tasks, but little research has been done about remote sensing. Tao et al. [43] proposed a self-supervised learning (SSL) method, and they demonstrated that the SSL paradigm outperforms the traditional methods. Moreover, they analyzed the impacts of three factors on RSI scene classification. They only made an attempt on small-scale datasets that the train images are less than 30,000. Zhao et al. [44] proposed a MTL framework integrating self-supervised learning method and scene classification tasks, so that the CNN models can train more discriminant features, but the number of parameters are not increasing. Kang et al. [45] introduced a new unsupervised deep metric learning model, the model integrated the advantages of spatial enhancement contrast loss and momentum update-based optimization. Although they conducted self-supervised learning task training on two datasets (NAIP dataset and Eurosat dataset) that include 100,000 images, respectively, they also used the same type of datasets in the downstream task to test the performance of the model, which did not take advantage of the strong generalization performance of pre-trained model from self-supervised learning. Stojni et al. [26] proposed a contrastive multi-view coding (CMC) self-supervised learning framework, which was trained in a natural scene dataset (ImageNet1000), high-resolution remote sensing datasets (NWPU-RESISC45 dataset, DOTA dataset, and NWPUVHR-10 dataset) and a low- resolution dataset (bigearthnet). The number of images they contained was 1,200,000, 196,215 and 269,695, respectively. Although the scale of remote sensing datasets they used is larger than before, high-resolution remote sensing datasets are still hand-crafted datasets. They did not use the labels, and the categories of scenes in the datasets are limited, which can not fully represent the diversity and complexity of remote sensing image scenes.

3. Proposed Method

The self-supervised learning network proposed in this paper is essentially an autoencoder structure, as shown in Figure 1. Rumelhart et al. [46] proposed the concept of Auto Encoder (AE) in 1986 and applied it to high-dimensional complex data processing, which promoted the development of neural networks. He et al. [47] proposed a masked autoencoder for self-supervised learning. AE is an unsupervised learning algorithm, and the network structure generally consists of two parts: an encoder for feature extraction and a decoder for target reconstruction. Generally speaking, the structure of the encoding and decoding of the autoencoder is symmetric, but the autoencoder framework designed in this paper is an asymmetric structure. The encoder can label the order of image patches by position embedding. In addition, it can also process the mask and mask token of the image patches, and the decoder does not contain these functions.

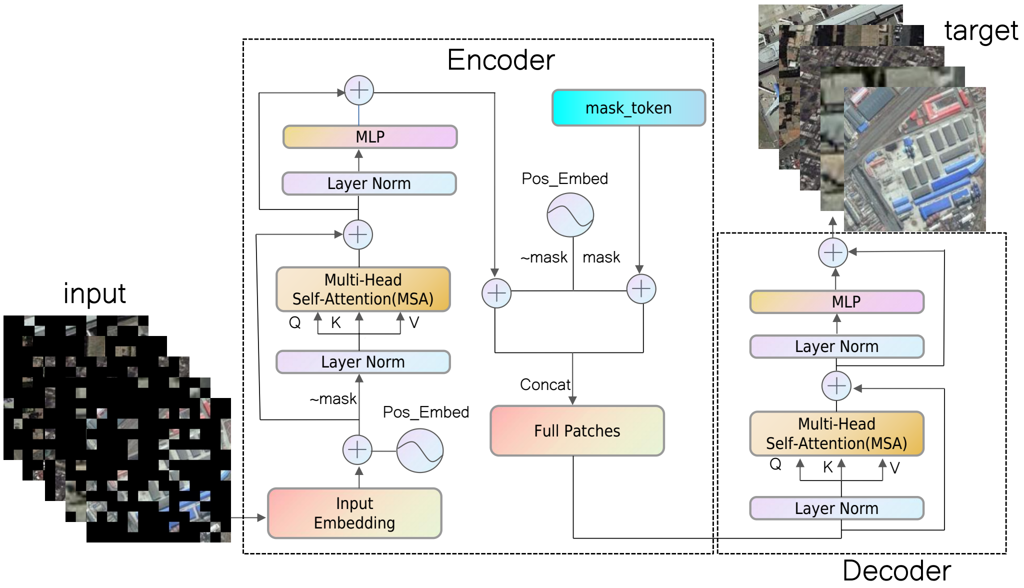


Figure 1. The overall architecture of the self-supervised learning network.

A. VIT

The Vision Transformer (VIT) model is developed based on the transformer model and is widely applied in computer vision. The Transformer was introduced by Vaswani et al. [48] to overcome the defects that recurrent neural networks cannot be parallelized in natural language processing (NLP), which consists of an encoder and a decoder. In the encoding stage, the words in the sentence are firstly converted into word vectors, then the global self-attention feature map is obtained through the self-attention module, residual connection, and layer normalization. Finally, the output of the encoder is obtained by the feedforward network, residual connection, and layer normalization. The VIT model proposed by Dosovitskiy et al. [49] is the first time that the encoder structure of the Transformer is directly used for image classification, laying a solid foundation for the development of the transformer in computer vision. To adapt to the structure of the input data of the transformer, the images need to be divided into non-overlapping image patches in the VIT model, and then the image patches are flattened and embedded in position encoding. Finally, a one-dimensional vector is obtained. The application of the transformer in computer vision usually uses this method to input images or feature maps. For the image classification task, a classifier is usually connected in the end to map the output features by the encoder to the scene category. In the process of scene classification, the useful information is usually concentrated in a specific area. The traditional CNN architecture is a filter based on a local receptive field, which treats each pixel equally. However, for the scene classification of the whole image, the contribution of each pixel in the image is different. For example, airports are mainly identified according to the aircraft contained in the image, while commercial and residential areas are identified according to the geometric outline, texture features, and spatial arrangement of buildings. VIT model adopts a multi-head self-attention (MSA) mechanism, which can automatically learn the contribution of each pixel to scene classification, which is more in line with the image recognition process of the real human visual system.

B. Encoder

As shown in Figure 1, the left side of the encoder is composed of the standard VIT structure, in which the classification header has been removed, and the function of the right part is to restore the number of image patches. During the data processing of the encoder, we discard the masked image patches. Here, we add the positional embeddings to the output features of VIT and mask tokens of the image, respectively, and concatenate them together, then the length of concatenated feature vectors is equal to the total number of image patches. Finally, the output features are sent to the decoder. The main structure of VIT is composed of an embedding layer and a transformer encoder. The transformer encoder cannot process two-dimensional images directly, and can only accept one-dimensional vectors as input. Therefore, when using VIT, it is necessary to map the two-dimensional matrix to the one-dimensional vector.

We suppose that the size of the input remote sensing image P is $c \times h \times w$, where c represents the number of channels, h represents the height, and w represents the width. We uniformly divide the image into non-overlapping image patches, and the size is $c \times h_p \times w_p$, h_p and w_p, usually set to 16, then the total number of image patches $m = hw/h_pw_p$. All the image patches are concatenated together to form a patch sequence $(P_1, P_2, \ldots, P_m)$ with a length of m, and then the patch sequence is input into the linear embedding layer and projected into a vector of dimension N, then the vector can be used as input data of the transformer encoder. The process of linear embedding can be expressed by the following formula:

$$L_0 = [P_1E; P_2E; \ldots; P_mE] + E_{pos}, E \in R^{(h_pw_pc)\times d}, \\ E_{pos} \in R^{(m+1)\times d} \tag{1}$$

where E represents a learnable embedding matrix and E_{pos} represents position encoding. Since the transformer's self-attention is disordered, the patch sequence $(P_1, P_2, \ldots, P_m)$ is regarded as a group of disordered patches. To maintain the relative spatial position of the patch in the raw image, E_{pos} is used to embed the position information into the patch sequence. $[P_1E; P_2E; \ldots; P_mE]$ represents that we concatenate the P_1E, P_2E, …, and P_mE together. The position encoding method adopted in this paper can be expressed by the following formula [50]:

$$E_{pos(2i)} = \sin(pos/10{,}000_{2i/d}) \\ E_{pos(2i+1)} = \cos(pos/10{,}000_{2i/d}) \tag{2}$$

where pos represents the regional position index of the image and d represents the position encoding dimension, i represents the ith dimension of the position encoding vector.

The paper adopts a random mask-based self-supervised learning framework to predict deleted patches from visible unmasked patches. The transformer encoder only processes the visible patches, so the masked patches are removed from the vector L_0 after the linear embedding operation. If the number of patches to be masked is n, then the number of visible patches is $m - n$. After this step, the patch sequence L_1 can be expressed by the following formula:

$$L_1 = [P_1; P_2; \ldots; P_{m-n}] \tag{3}$$

$[P_1; P_2; \ldots, P_{m-n}]$ represents that we concatenate the P_1, P_2, …, and P_{m-n} together. The transformer encoder takes L_1 as the input sequence for feature extraction. The Transformer encoder consists of several layers with same structure. Each layer mainly includes a MSA module and a multi-layer perceptron (MLP) module with. To ensure the consistency of data distribution, the data must be processed by a normalization layer before being input into each module. In addition, a residual skip connection is used in each module. The perceptron layer consists of two fully connected layers, then nonlinear mapping is performed between the two dense layers through a GeLU activation function. The transformer encoder is made up of several such units. If the number of such units is k, the calculation process of the transformer encoder can be expressed as following:

$$L'_k = MSA(LN(L_k - 1)) + L_k - 1, k = 1, 2, \ldots, K \\ L_k = MSA(LN(L'_k)) + L'_k, k = 1, 2, \ldots, K \tag{4}$$

where L_k represents the kth layer of the transformer encoder, LN represents the abbreviation of Layer Norm, and MSA represents the abbreviation of MSA.

The last layer of the encoder is to integrate the output of the features by the transformer encoder and the features of the masked patches while embedding their position information separately. Finally, they are concatenated and sent to the decoder for the reconstruction of the discarded image patches. The masked patches feature can be represented by a trainable and shared mask token. The whole process can be expressed as follows:

$$\begin{gathered} T = [p'_1; p'_2; \ldots; p'_n] + E^M_{pos} \\ L_{cat} = [L_k + E^V_{pos}; T] \end{gathered} \tag{5}$$

In the above formula, p'_n represents the nth mask token, E^M_{pos} represents the position embedded feature of the image patches that are masked, L_k represents the feature vector output by the Transformer encoder, and E^V_{pos} represents the position embedded feature of the image patches that are not masked. L_{cat} represents that the mask token of the embedded position information and the previously extracted features are concatenated together, and the length of the features vector is exactly equal to the number of all image patches.

C. MSA mechanism

MSA is a key part of VIT, and the structure is shown in Figure 2. Firstly, three learnable matrices that represent different weights are used to transform the input matrix of the encoder to obtain the query matrix Q, the key matrix K, and the value matrix V. Then, through the scaled dot-product attention, the self-attention feature map is calculated. The ultimate goal of multi-head self-attention is to obtain multi-independent attention feature maps, which are realized by multi-group transformation matrices. Finally, the multi-head attention feature map is obtained by concatenating different attention feature maps. The formula for calculating attention features is as follows

$$Attention(Q, K, V) = softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)V \tag{6}$$

Among them, Q, K, and V are two-dimensional matrices composed of vectors, the dot product between the matrix Q and the transpose of the matrix K represents a correlation matrix. Because Q and K come from the transformation of the same matrix, the correlation matrix can describe the correlation between the input vectors. To avoid the vanishing gradients caused by the softmax operation, a coefficient is used to scale the correlation matrix. $\sqrt{d_k}$ represents a coefficient, where d_k represents the dimension of the vector k, which is used to scale the output. The activated correlation matrix is dot-multiplied with V to obtain the global self-attention map.

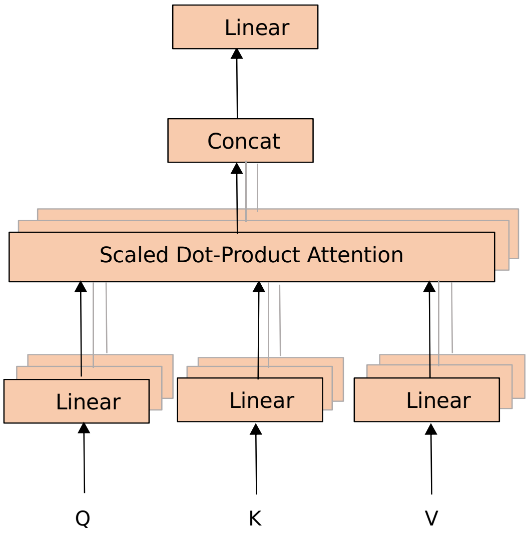


Figure 2. Multi-headed self-attention.

D. Decoder

The main structure of the decoder is the same as that of the transformer encoder in the encoder, which consists of a stack of MSA modules and MLP modules. The role of the decoder is to recover the masked image patches by predicting a vector of pixel values for each masked token. The fully connected layer in the end is used to map the number of feature channels output by the transformer encoder to be consistent with the number of pixels in the masked image patches. Finally, the predicted mask patch pixel value is filled to its original position, thus completing the reconstruction of the entire image. It is worth emphasizing that in the training process, our loss function only calculates the mean square error of the predicted pixels value and the original image pixels value for the masked image patches because we only predict the pixels value of the mask token.

E. Remote Sensing Image Scene Classification

When the training of the self-supervised learning model has been completed, we use the VIT module in the encoder to extract the features and add a classifier in the end of the network, they form a scene classifier together. The dimension of the output feature is equal to the scene categories. Although the structure of the scene classifier is similar to the VIT module in the encoder, there lies a significant difference between them. In the pre-training stage, the VIT module only needs to process the image patches that are not masked, but all the image patches need to be processed in the classifier, and the masked image patches are no longer removed.

4. Experiments and Analysis

A. Datasets description

In this experiment, we used 11 kinds of different datasets, each of which is described in detail in Table 1. Among them, UC Merced, WHU-RS19, SIRI-WHU, RSI-CB256, RSSCN7, RSC11, OPTIMAL-31, PatternNet, and GID datasets are used for pre-training of self-supervised learning models. The first 8 data sets are public remote sensing scene classification data sets with labels, with a total of 65,797 images. In the pre-training process, labels are not required. The image size is uniformly resized to 224 × 224. We name it the public image dataset (PID). Although unlabeled scene classification datasets have also been used in previous studies to train self-supervised models, these datasets are manually produced and carefully selected to identify specific kinds of scenes. The categories of scenes included are very limited, the number of samples is not large, and the production cost is very expensive. In a word, the unlabeled datasets are not naturally generated. GID dataset (Gaofen image dataset) is a remote sensing image's semantic segmentation dataset produced by Wuhan University, which contains 150 high-resolution Gaofen-2 images (the image size 6800 × 7200) obtained from more than 60 cities in China. These images cover a geographical area of more than 50,000 square kilometers. Images in GID have high intra-class similarity and low inter-class separability. We select images from which the distribution of ground objects is relatively uniform as training samples. Because the uneven samples will lead to a large number of redundant datasets. For example, most areas of the whole image are covered by water, farmland, and forests with single texture features, which means that the distribution of ground objects is uneven. Too many samples like these images will not help improve classification performance, but will lead to an increase in calculation cost. For training samples, each image is cut into non-overlapping image patches with the size of 224 × 224. In order to keep consistent with the number of images selected from public remote sensing scene classification data sets, we randomly selected 65,797 images from the generated image patches for the pre-training process of the self-supervised learning model. We combine PID and GID to construct a large-scale unlabeled dataset with a total of 131,594 images, which we name as Large Scale Image Dataset (LSID). In addition, to verify the performance of the self-supervised learning method, we selected two complex scene classification datasets as train datasets and test datasets for scene classification tasks, namely the NWPU-RESISC45(NWPU) and AID datasets, the detailed descriptions are shown in Table 1.

Table 1. The experiment datasets.

Datasets	Image Number Perclass	Number of Scene Categories	Total Number	Size	Data Sources
UC Merced [13]	100	21	2100	256 × 256	Aerial orthoimagery
WHU-RS19 [50]	50∼61	19	1005	600 × 600	Google Earth
SIRI-WHU [51]	200	12	2400	200 × 200	Google Earth
RSI-CB256 [52]	about 690	35	24,000	256 × 256	Google Earth and Bing Maps
RSSCN7 [53]	400	7	2800	400 × 400	Google Earth
RSC11 [54]	about 100	11	1232	512 × 512	Google Earth
OPTIMAL-31 [55]	60	31	1860	256 × 256	Google Earth
PatternNet [56]	800	38	30,400	256 × 256	Google Earth
GID [57]	-	-	-	6800 × 7200	Gaofen-2
NWPU-RESISC45 [4]	700	45	31,500	256 × 256	Google Earth
AID [58]	220∼420	30	10,000	600 × 600	Google Earth

B. Experiment Setup

Our experiments are composed of two groups. The purpose of the first group is to verify the performance of our proposed self-supervised learning network. We conduct pre-training experiments on the LSID dataset, and then perform the scene classification experiment on the NWPU dataset and AID dataset, respectively, which we call self-supervised learning scene classification (SSLSC). Meanwhile, we also exploit the VIT model pre-trained on the natural dataset to conduct scene classification experiments on the NWPU dataset and AID dataset to compare and analyze the advantages of the SSLSC method. We also use the VIT model for scene classification based on the SSLSC method, which has the same structure as the VIT model used for pre-training on natural datasets. The other group of the experiment is designed to compare the performance of our proposed self-supervised learning network on the PID and AID datasets, the models are pre-trained on these two datasets, respectively. Then, we fine-tune the model on the NWPU dataset and AID dataset for scene classification. In the pre-training stage of the SSLSC method, we set the learning rate to 1.5×10^{-4}, the epoch to 500, and the batch to 64. The depth of the transformer encoder in the encoder is set to 12, and it is set to 4 in the decoder. The random mask ratio of the image patches is set to 0.75, which means 75% of the image patches are randomly discarded for each image during the encoding stage and then reconstructed during the decoding stage. In the model fine-tuning stage of the SSLSC method, the learning rate is set to 1×10^{-3}, the epoch is set to 200, the batch is set to 64, and the depth of the transformer encoder in the encoder is set to 12. The attention head is set to 12 and the embedding layer dimension is set to 768 in all encoders. For the NWPU data set, the training ratio is set to 10% and 20%, whereas for the AID data set, the training ratio is set to 20% and 50%, respectively. To obtain reliable results, the experiments were repeated 10 times during scene classification. We choose the overall accuracy and confusion matrix to evaluate the performance of the model and finally calculate the average of the overall accuracy of 10 experiments as the accuracy of the model. In addition, we use the best model in the training process to calculate the confusion matrix. The settings of these parameters of the VIT model pre-trained on the natural dataset are consistent with the VIT model fine-tuned by SSLSC, and we choose the ViT-B_16 model trained on the ImageNet2021 and ImageNet-21 data sets as the pre-trained model. All our experiments are conducted on a computer, which has 2.70 GHz × 12 core CPU with 32GB memory, configured with a GeForce GTX 1080 Ti graphics card with 11GB memory capacity.

C. Experimental results of SSLSC

To validate the effectiveness of our proposed method, we compare it with some other state-of-the-art methods in this section. As is known to all, the performance of the method based on deep learning in scene classification far surpasses the traditional machine learning method based on handcrafted feature extraction. Therefore, the methods used for comparison are all based on deep learning.

(1) Result on LSID dataset: LSID is a large-scale unlabeled training dataset that we constructed to train the self-supervised learning task. The goal of this stage is to learn the features of the image from the unmasked image patches, so as to recover the pixels of the masked image patches. The final pre-trained model can be used as a feature extractor for fine-tuning in downstream scene classification tasks. We can show the reconstruction effect of the model on the pixels of the masked image patches. The better the reconstruction effect of image pixels, the more representative the features extracted by the encoder of the self-supervised learning framework, and it also shows that the pre-trained model is suitable as a feature extractor for downstream tasks. Now, we randomly select an image from each of five typical scene categories which represent mountain, dense residential, commercial area, church, and airport to restore the image pixels and evaluate the feature extraction ability of the proposed model. We carried out the experiments on the NWPU dataset and AID dataset, respectively.

The experimental results on the two datasets are shown in Figures 3 and 4, respectively. The images of five scene categories are represented from left to right, which are mountain, dense residential, commercial area, church, and airport. The images from top to bottom represent the original image, the masked image, and the reconstructed image in turn. Comparing the results in Figures 3 and 4, we can find that the discarded pixels can be well recovered. In particular, through the comparison between the original image and the reconstructed image, we can see that the latter can be consistent with the former in color and texture. Through analysis, we can draw a conclusion that the encoder of the self-supervised learning framework proposed in this paper can extract representative image features, which have good robustness. The pre-trained model can well predict the discarded pixels of the masked image patches on the NWPU dataset and AID dataset.

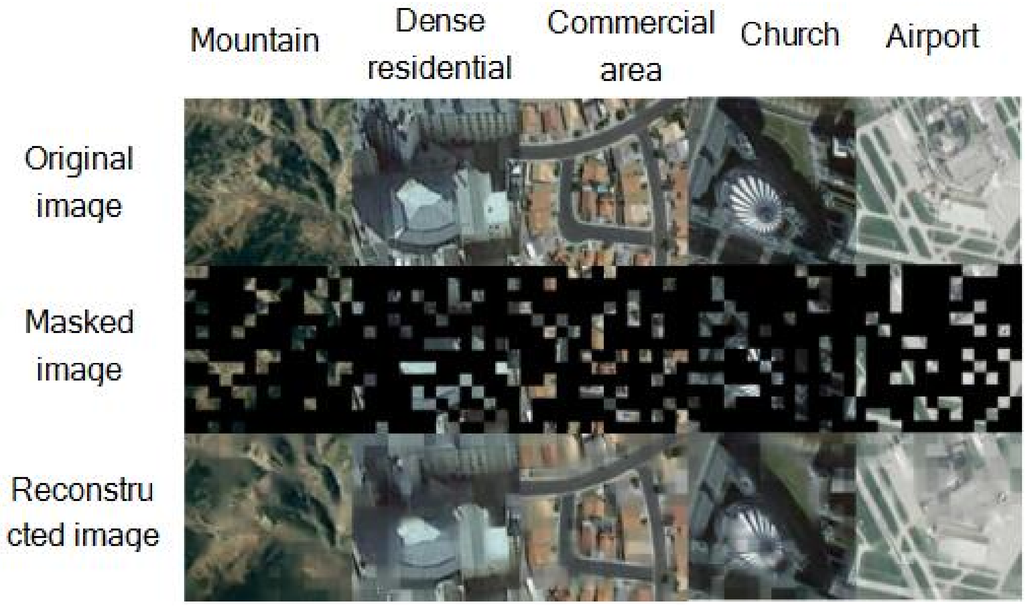


Figure 3. Pixel reconstruction results of self-supervised learning pre-trained model on NWPU data set.

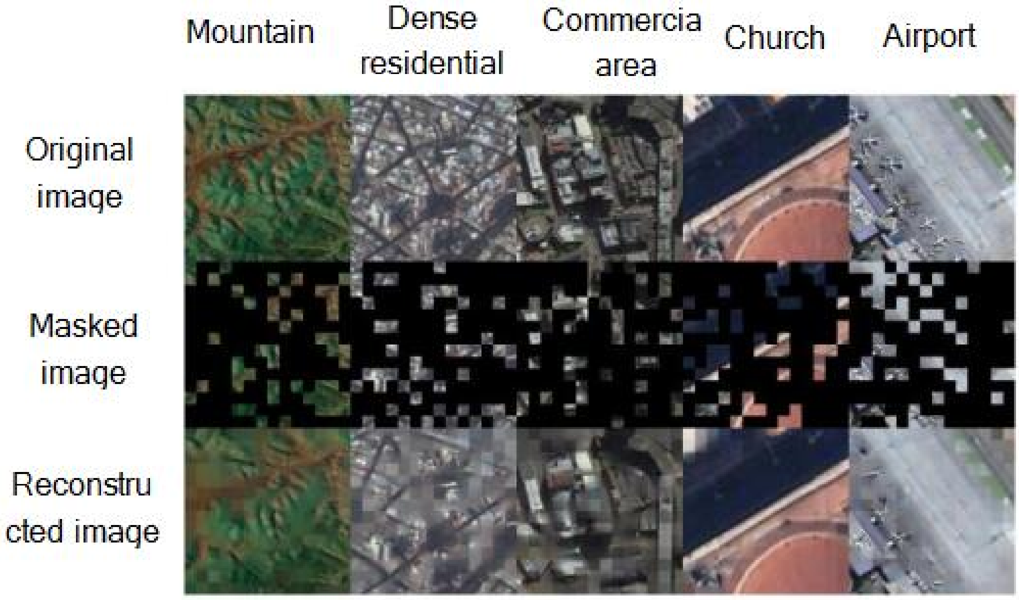


Figure 4. Pixel reconstruction results of self-supervised learning pre-trained model on AID data set.

(2) Result on NWPU dataset: Table 2 shows the overall accuracy comparison of scene classification, and Figure 5 shows the classification confusion matrix, all of which are based on the NWPU dataset. The methods in Table 2 mainly include the current mainstream CNN-based methods and their variants. For example, AlexNet [20], GoogleNet [22], VGGNet-16 [21] and EfficientNet [59] are all classic CNN networks. D-CNN with AlexNet [60], D-CNN with GoogleNet [60], D-CNN with VGGNet-16 [60], Siamese ResNet50 [61] and ResNeXt-101+MTL [44] are variants of AlexNet, GoogleNet, VGGNet-16, ResNet50 and ResNeXt-101, respectively. In addition, we also list some excellent methods proposed in the past two years and compare their classification accuracies. These methods include RCOVBOVW [62], EMTCAL [63], T-CNN [64], and ResNet-101+EAM [65]. From Table 2, it can be observed that the overall accuracy of our method surpasses all methods. Among the previous methods listed in the table, Hydra [66] has the best performance on the 10% training dataset, with an overall accuracy of 92.44%, but our method has an overall accuracy of 92.68%, which is 0.24% higher than the former. Hydra [66] is also the method that performs best on 20% of the training dataset, the overall accuracy reaches 94.51%, but our method has an overall accuracy of 94.73%, which is 0.22% higher than the former. It is worth emphasizing that ResNeXt-101+MTL [44] is also a self-supervised learning method, and on 10% and 20% training datasets, the overall accuracy is 91.91% and 94.21%, respectively. Compared with this, our overall accuracy is improved by 0.77% and 0.52%, respectively. In addition, ViT-B_16 [224 × 224] represents the Vision Transformer framework, the input image size is 224 × 224, and the image is divided into 16 × 16 image patches. We use the pre-trained model, which is trained on two natural datasets (ImageNet2021 and ImageNet-21), to fine-tune for the scene classification task, the overall accuracy is 91.89% on 10% of the training dataset, and which is 93.26% on 20% of that dataset. It is obviously that our overall accuracy is improved by 0.79% and 1.47%, respectively.

Table 2. OA (%) of eighteen kinds of methods and our method under the training ratio of 10% and 20% on NWPU dataset.

Method	**Training Ratio**	
	10%	**20%**
AlexNet [20]	76.69 ± 0.21	79.85 ± 0.13
GoogleNet [22]	76.19 ± 0.38	78.48 ± 0.26
VGGNet-16 [21]	76.47 ± 0.18	79.79 ± 0.15
D-CNN based on AlexNet [60]	85.56 ± 0.20	87.24 ± 0.12
D-CNN based on GoogleNet [60]	86.89 ± 0.10	90.49 ± 0.15
D-CNN based on VGGNet-16 [60]	89.22 ± 0.5	91.89 ± 0.22
Siamese ResNet50 [61]	-	92.28
EfficientNet [59]	90.91 ± 0.16	94.37 ± 0.14
ADSSM [67]	91.69 ± 0.22	94.29 ± 0.14
SF-CNN [68]	89.89 ± 0.16	92.55 ± 0.14
DNE [69]	-	96.01
Hydra [66]	92.44 ± 0.34	94.51 ± 0.21
ResNeXt-101+MTL [44]	91.91 ± 0.18	94.21 ± 0.15
RCOVBOVW [62]	90.25 ± 0.13	92.37 ± 0.18
EMTCAL [63]	91.63 ± 0.19	93.65 ± 0.12
T-CNN [64]	90.25 ± 0.14	93.05 ± 0.12
ResNet-101+EAM [65]	91.91 ± 0.12	94.29 ± 0.09
ViT-B_16 [224 × 224]	91.89 ± 0.43	93.26 ± 0.1
SSLSC+LSID	92.68 ± 0.15	94.73 ± 0.11

In addition, Figure 5 shows the confusion matrix for scene classification on the NWPU dataset. From Figure 5, we can observe that the proposed model can identify most of the scene categories correctly, their overall accuracy exceeds 90%, especially for the images with scene categories of airplane, chaparral, circular farmland, cloud, harbor, parking lot, snowberg, and storage tank, the overall accuracy reaches 100%. However, the recognition accuracy is relatively low for some scene categories. For example, for church, commercial area, mountain, palace, railway station, and wetland, their overall accuracies are 87%, 89%, 87%, 84%, 89% and 89%, respectively. Through further analysis, we can summarize the reasons for the lower recognition accuracy of these scene categories. Figure 5 shows that 8% of the churches were mistakenly classified as palaces, while 11% of the palaces were mistakenly classified as churches, because churches and palaces have many similarities in architectural structure. Overall, 6% of the mountains were identified as deserts because the sparsely vegetated mountains had a similar color to the desert, and 7% of the railway stations are identified as railways. According to common sense, there are generally dense railways around the railway station, and they are easy to identify as railways. Additionally, 6% of wetlands were identified as lakes because both wetlands and lakes contain water bodies, and wetlands are easily classified as lakes when there is a lot of water in them.

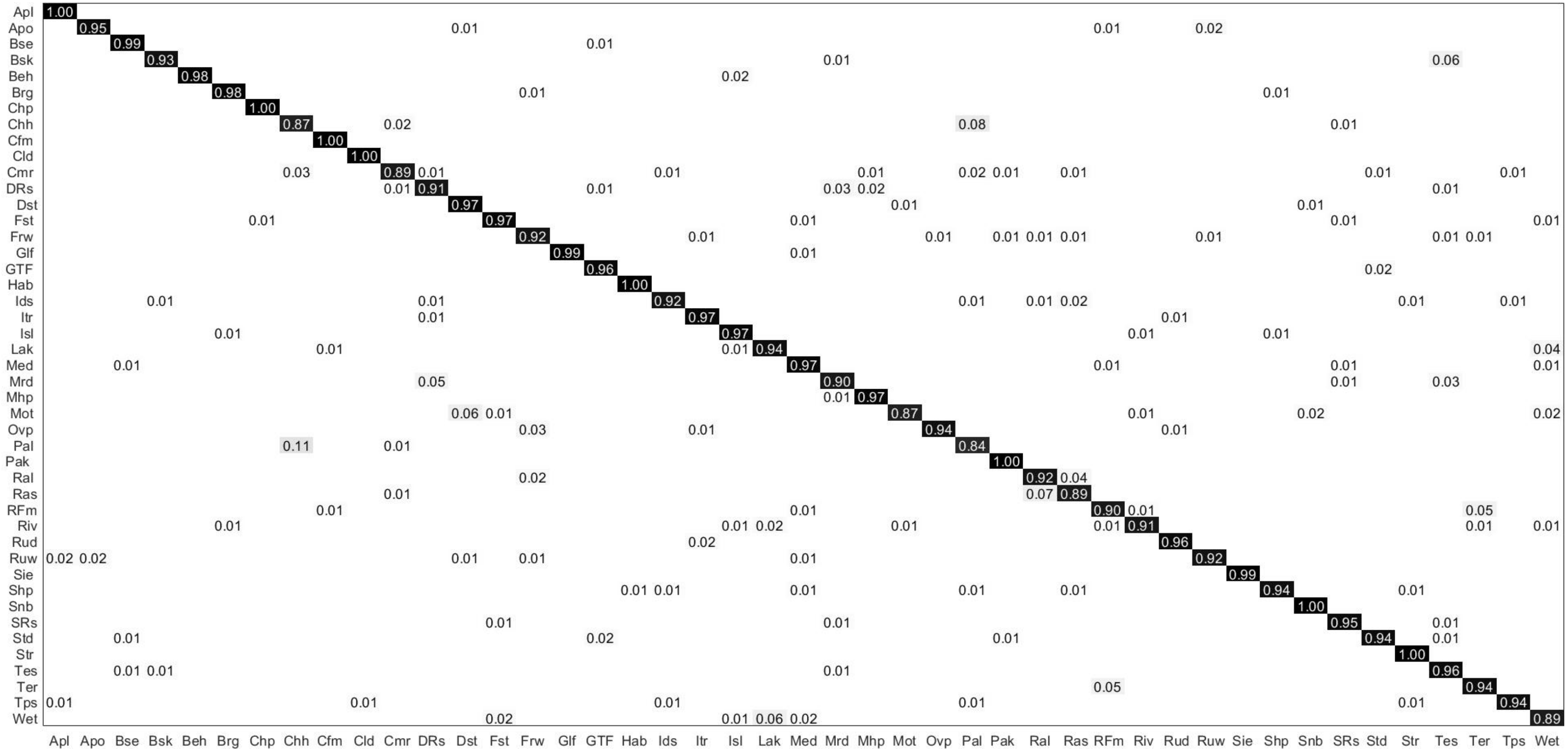


Figure 5. Confusion matrix of our proposed method on NWPU dataset.

(3) Result on AID dataset: Table 3 and Figure 6 are the scene classification results on the AID dataset. Similar to Table 2, the methods compared in Table 3 mainly include the current mainstream CNN-based methods and their variants. We can see that our method has the highest overall accuracy than other methods when the training ratio is set to 50%. The accuracies are 94.13% and 97.44% on the 20% and 50% training datasets, respectively. In the previous methods, on 20% training dataset, the method with the highest accuracy is EMTCAL [63], and the overall accuracy reaches 94.69%. Unfortunately, the accuracy of our method is 0.56% lower than that. On the 50% training dataset, the method with the highest accuracy is CNNs-WD [70], and the accuracy reaches 97.24%. Although CNNs-WD [70] is already excellent, the accuracy is 0.46% lower than that of our method. In addition, we also compare with other self-supervised learning methods and Vision Transformer, among which, CMC [26] and ResNeXt-101+MTL [44] are self-supervised learning methods. On the 50% training dataset, the overall accuracy of CMC [26] is 95.58%, and the overall accuracy of ResNeXt-101+MTL [44] is 96.89%, but our method outperforms them by 1.86% and 0.55%, respectively. On two different rates of a training dataset, the overall accuracy of Vision Transformer is 93.54% and 96.25%, which are 0.59% and 1.19% lower than our method. Figure 6 shows the confusion matrix. It can be observed from Figure 6 that nearly half of the scene categories have an overall accuracy of 100%. They are baseball field, beach, bridge, desert, forest, meadow, mountain, pond, port, river, sparse residential, stadium, storage tanks, and viaducts. Only two scene categories have overall accuracies below 90%, resort and square, with accuracies of 87% and 89%, respectively. Through further analysis, it can be seen that 3% of the resorts are identified as parks, and 3% of the resorts are identified as schools. In addition, 3% of squares are identified as center, and 3% of squares are identified as church. Resort, park, and square are all interspersed with trees and grass among the sparse buildings, there are some similarities in spectral and spatial arrangement. Square, center, and church have a common feature that they all contain circular building structures, then the geometric shapes shown in the images have certain similarities, so these scene categories are very easy to be misclassified. Compared with the current state-of-the-art methods, this shows that our method is superior to most of the previous methods.

D. Comparison of Classification Accuracies Based on Different Pre-Training Datasets

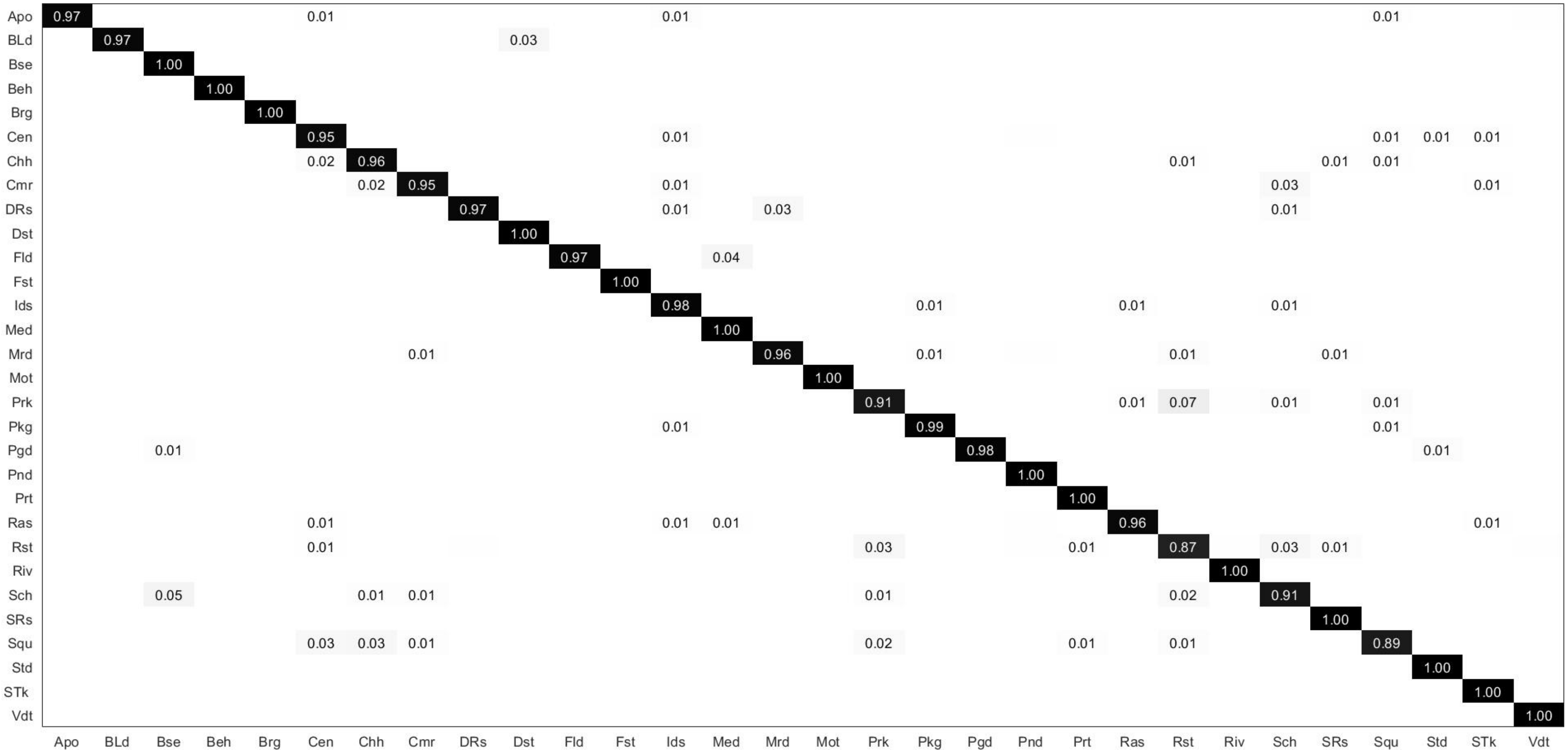
As described in the dataset description section above, the GID dataset is an unlabeled data set constructed by directly cropping the visible light band of the Gaofen-2 remote sensing image into image patches. The generation process requires no manual intervention and can be completed in just a few minutes. The PID dataset is produced by integrating many public scene classification datasets. Although we have not used labels, the construction process of these datasets is very complex, and they are produced by careful manual selection. Labeling datasets requires strong professional knowledge and rich experience, and it also takes a long time. Therefore, so far, there are not enough labeled datasets that can be used for scene classification in remote sensing. In this section, we use GID and PID datasets to train the self-supervised learning model, respectively, then fine-tune the pre-trained model on the labeled scene classification dataset, and evaluate the performance of the model with overall accuracy. Our experiments are performed on the NWPU and AID datasets, respectively. Additionally, we also compare the results with ViT-B_16 [224 × 224] and SSLSC+LSID. Tables 4 and 5 show the experimental results. It can be seen from Table 4 that the overall accuracy of SSLSC+PID is higher than that of SSLSC+GID, which is 0.24% higher on 10% training set and 0.4% higher on 20% training set respectively. Compared with SSLSC+LSID, the overall accuracy of SSLSC+PID is 5.23% and 3% lower on the 10% and 20% training sets, respectively. Table 5 shows the results on the AID dataset. The overall accuracy of SSLSC+PID is 0.46% higher than that of SSLSC+GID on 20% of the training rate. However, on 50% training rate, the results are exactly the opposite. The overall accuracy of SSLSC+PID is 0.2% higher than that of SSLSC+GID. Compared with SSLSC+LSID, the overall accuracy of SSLSC+PID is 2.39% and 2.54% lower on the 20% and 50% training sets, respectively. In addition, both the metrics of SSLSC+GID and SSLSC+PID in Tables 4 and 5 were lower than ViT-B_16 [224 × 224]. The ViT-B_16 [224 × 224] is a traditional supervised learning framework. In this experiment, we choose the model trained on ImageNet2021 and ImageNet-21 datasets, which have more than one million labeled samples and contain 1000 scene categories as the pretrained model. However, in the comparison experiments, both PID and GID datasets contain 65,797 samples, and compared with natural image datasets, these two datasets are much smaller in scale. Hence, the overall accuracy of ViT-B_16 [224 × 224] is lower. The LSID dataset is composed of the PID dataset and the GID dataset. In all combinations in Tables 4 and 5, the performance of SSLSC+LSID outperforms the other methods.

Table 3. OA (%) of twenty kinds of methods and our method under the training ratio of 20% and 50% on AID dataset.

Method	**Training Ratio**	
	20%	**50%**
CaffeNet [58]	86.86 ± 0.47	89.53 ± 0.31
VGG-VD-16 [58]	86.59 ± 0.29	89.64 ± 0.36
GoogLeNet [58]	83.44 ± 0.40	86.39 ± 0.55
DenseNet121 [71]	93.76 ± 0.23	94.73 ± 0.26
DenseNet169 [71]	92.43 ± 0.36	94.17 ± 0.33
VGG19 [72]	87.73 ± 0.25	91.71 ± 0.42
ResNet50 [72]	92.39 ± 0.15	94.69 ± 0.19
InceptionV3 [72]	93.27 ± 0.17	95.07 ± 0.22
HW-CNNs [73]	-	96.98 ± 0.33
SF-CNN [68]	93.60 ± 0.12	96.66 ± 0.11
CNNs-WD [70]	-	97.24 ± 0.32
RSFJR [74]	-	96.81 ± 1.36
GBN [75]	92.20 ± 0.23	95.48 ± 0.12
CNN-CapsNet [76]	93.79 ± 0.13	96.32 ± 0.12
CMC [26]	-	95.58
ResNeXt-101+MTL [44]	93.96 ± 0.11	96.89 ± 0.18
EMTCAL [63]	94.69 ± 0.14	96.41 ± 0.23
T-CNN [64]	94.55 ± 0.27	96.72 ± 0.23
ResNet-101+EAM [65]	94.26 ± 0.11	97.06 ± 0.19
ViT-B_16 [224 × 224]	93.54 ± 0.25	96.25 ± 0.17
SSLSC+LSID	94.13 ± 0.29	97.44 ± 0.14

Table 4. OA (%) of experiment on NWPU with different pre-training datasets.

Method	**Training Ratio**	
	10%	**20%**
SSLSC+GID	87.21	91.33
SSLSC+PID	87.45	91.73
ViT-B_16 [224 × 224]	91.89	93.26
SSLSC+LSID	92.68	94.73



	Apo	BLd	Bse	Beh	Brg	Cen	Chh	Cmr	DRs	Dst	Fld	Fst	Ids	Med	Mrd	Mot	Prk	Pkg	Pgd	Pnd	Prt	Ras	Rst	Riv	Sch	SRs	Squ	Std	STk	Vdt
Apo	0.97					0.01							0.01														0.01			
BLd		0.97								0.03																				
Bse			1.00																											
Beh				1.00																										
Brg					1.00																									
Cen						0.95							0.01														0.01	0.01	0.01	
Chh						0.02	0.96																0.01			0.01	0.01			
Cmr							0.02	0.95					0.01												0.03				0.01	
DRs									0.97				0.01		0.03										0.01					
Dst										1.00																				
Fld											0.97			0.04																
Fst												1.00																		
Ids													0.98					0.01				0.01			0.01					
Med														1.00																
Mrd								0.01							0.96			0.01					0.01			0.01				
Mot																1.00														
Prk																	0.91					0.01	0.07		0.01		0.01			
Pkg													0.01					0.99									0.01			
Pgd			0.01																0.98									0.01		
Pnd																				1.00										
Prt																					1.00									
Ras						0.01							0.01	0.01								0.96							0.01	
Rst						0.01											0.03				0.01		0.87		0.03	0.01				
Riv																								1.00						
Sch			0.05				0.01	0.01									0.01						0.02		0.91					
SRs																										1.00				
Squ						0.03	0.03	0.01									0.02				0.01		0.01				0.89			
Std																												1.00		
STk																													1.00	
Vdt																														1.00

Figure 6. Confusion matrix of our proposed method on AID dataset.

Table 5. OA (%) of experiment on AID with different pre-training datasets.

Method	Training Ratio	
	20%	**50%**
SSLSC+GID	91.28	95.10
SSLSC+PID	91.74	94.90
ViT-B_16 [224 × 224]	93.54	96.25
SSLSC+LSID	94.13	97.44

Through further analysis, it can be found that the PID dataset has several advantages over the GID dataset. First of all, the PID dataset has high inter-class dissimilarity and low inter-class similarity. Because it is manually selected, the distribution of ground objects in each image is more in line with the corresponding scene category. Secondly, for the downstream scene classification task, the distribution of scene categories in the PID dataset is more uniform, which can ensure that each scene category has a certain amount of data. Taking churches as an example, the GID data comes from cities in China, where the number of churches is relatively small. Finally, the PID dataset comes from a wide range of sources, some from aerial orthophotos, some from Google Earth, etc., which included different types of sensors and different sizes of spatial resolution. Therefore, the overall accuracy on the PID dataset is slightly higher in most cases. In general, the accuracy difference between SSLSC+GID and SSLSC+PID is not great, but the GID dataset is very easy to obtain, which has great advantages in applying large-scale unlabeled datasets to the training of self-supervised learning.

E. Effects of different pre-training models on training loss

In this section, we mainly analyze the effects of different pre-training models on training loss. LSID, GID, and PID represent three different pre-training datasets, respectively. We pre-train the self-supervised model on these three datasets to form three different kinds of pre-trained models. Then, we fine-tune the models on the downstream task of remote sensing scene classification; SSLSC + LSID, SSLSC + GID, and SSLSC + PID represent these three different methods, respectively. VIT stands for Vision Transformer, which uses a model (i.e., ViT-B_16 [224 × 224]) pre-trained on a natural dataset. We analyze the effects of different pre-training models on training loss in the process of training the scene classification model. Our experiments are performed on the NWPU dataset and the AID dataset, the rate of the training set is set to 20% and 50%, respectively, and the epoch is set to 200. The results are shown in Figures 7 and 8. We can observe from Figures 7 and 8 that the changing trend of the loss value of SSLSC+LSID, SSLSC+GID, and SSLSC+PID on the two data sets is almost the same, and the loss value decreases rapidly with the progress of training. When the epoch is greater than 75, the loss value only fluctuates in a small range, and the model has converged. However, compared with our method, the changing trend of the loss value of VIT is quite different. On the NWPU dataset, the loss value tends to stabilize when the epoch value is about 150. On the AID dataset, the model starts to converge when the epoch value is greater than 170. In addition, the variation range of losses is also quite different on two different pre-trained models. On the pre-training model based on unlabeled remote sensing data sets, the loss range is smaller, between 1.5 and 4. On pre-trained models based on natural data sets, the loss variation is much larger, between 0 and 4. Which may be one of the reasons why our method can reach convergence first. In general, compared with the VIT model, which is pre-trained on the natural dataset, our method can reach convergence first, which can save training time. Further analysis shows that an unlabeled remote sensing dataset we used in the pre-trained stage is consistent with the scene classification dataset in the downstream task in terms of spectrum, texture, and geometric structure. Therefore, our method can learn more representative image representations, convergence can be achieved with fewer iterations. In addition, the dataset we used does not need manual annotation, so we can complete the task of scene classification on remote sensing image datasets more efficiently.

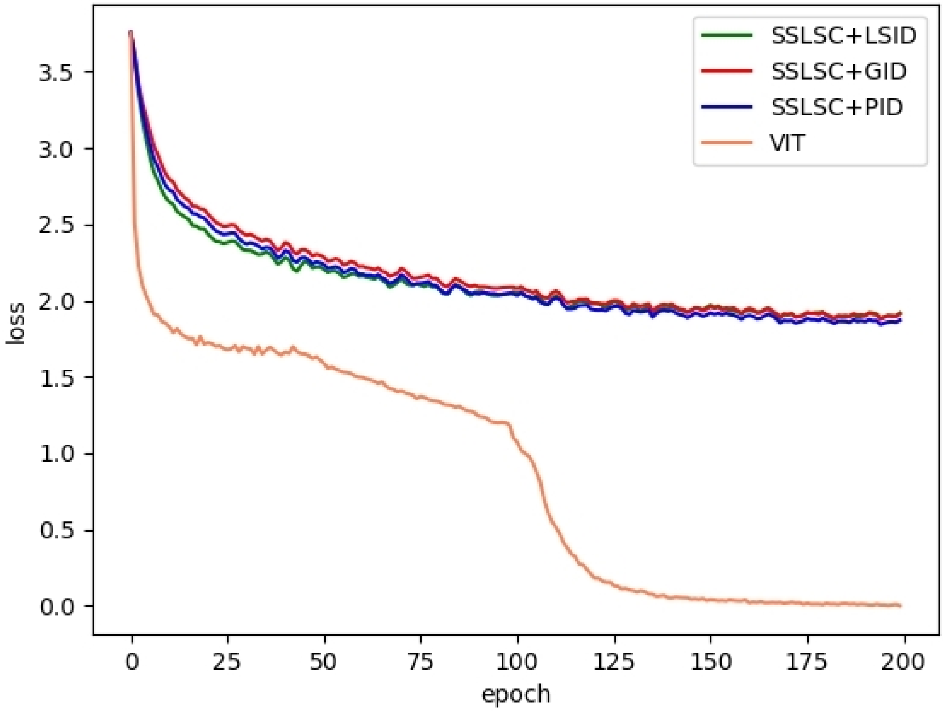


Figure 7. Train loss of our proposed method on NWPU dataset.

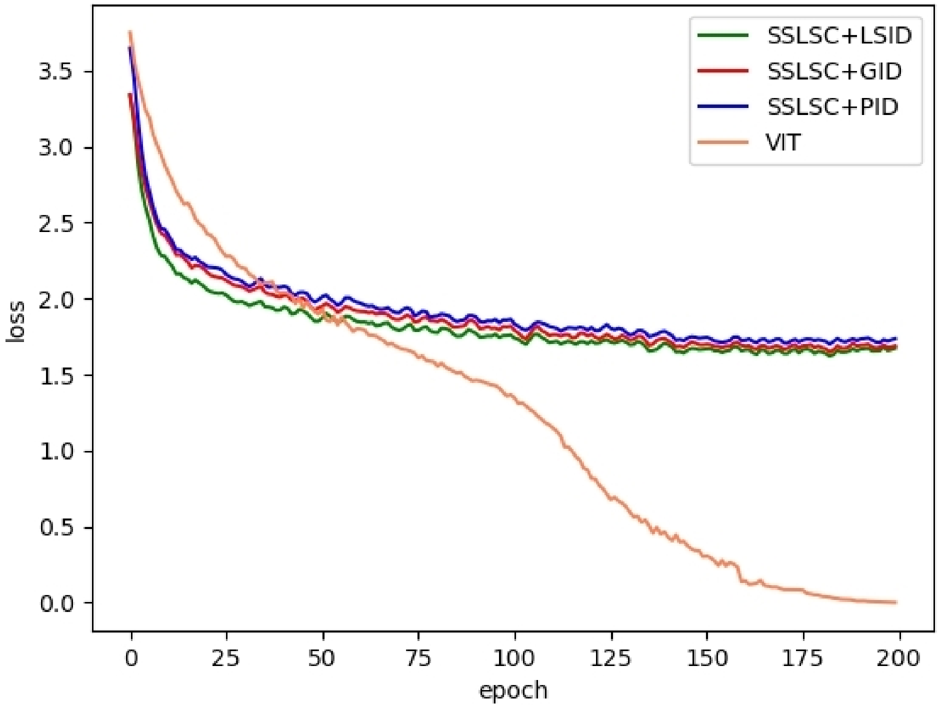


Figure 8. Train loss of our proposed method on AID dataset.

5. Conclusions

In this study, we proposed a scene classification method for a remote sensing image based on a self-supervised learning framework, which can use an unlabeled dataset to learn the representation of the image, and then transfer the learned knowledge to the scene classification task. The self-supervised learning framework consists of an encoder and a decoder. The function of the encoder is to process the unmasked image patches, learn the feature representation of the image, and then combine them with the mask token of the image to form a joint encoding. The role of the decoder is to reconstruct the pixels of the masked image patches. The whole process is trained on unlabeled images, and the training dataset is constructed based on Gaofen-2 remote sensing images and public scene classification datasets. Then, we transfer the parameter weights learned by the encoder to fine-tune the scene classification task. The scene classification experiments were carried out on two public and challenging datasets (NWPU dataset and AID dataset), respectively. We used the OA of the classification and the confusion matrix to evaluate the performance of the model and compared it with the current mainstream scene classification methods. We can draw a conclusion that our method is superior to most state-of-the-art methods. We conducted comparative experiments on two different unlabeled datasets, one of which was integrated based on the public hand-crafted scene classification datasets, and the other was randomly generated based on Gaofen-2 satellite imagery. Through analysis, our method can also achieve good performance on an unlabeled dataset that was generated without human intervention. Finally, we analyzed the effects of different pre-training models on training loss. The results show that our method can reach convergence with fewer iterations, then we can save the training time. In conclusion, our method can overcome the dependence on large-scale labeled datasets of the traditional supervised learning methods. The above experiments prove that our method has obvious advantages over other methods

In future research, we will consider using multi-source data to randomly generate unlabeled datasets to train the self-supervised learning model and then apply it to the scene classification of remote sensing images. These datasets include different spatial resolutions, different band combinations, different sensors, and different countries and regions, so as to further improve the accuracy of scene classification.

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

Funding: This research was funded by the National Natural Science Foundation of China under Grant 41871296.

Data Availability Statement: Not applicable.

Acknowledgments: We would like to express our gratitude to the editors and reviewers for carefully reviewing our manuscript and giving valuable comments.

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

References

1. Hu, F.; Xia, G.; Hu, J.; Zhang L. Transferring deep convolutional neural networks for the scene classification of high-resolution remote sensing imagery. *Remote Sens.* **2015**, *7*, 14680–14707. [CrossRef]
2. GU, Y.; Wang, Y.; Li, Y. A survey on deep learning-driven remote sensing image scene understanding: Scene classification, scene retrieval and scene-guided object detection. *Appl. Sci.* **2019**, *9*, 2110. [CrossRef]
3. Cheng, G.; Han, J. A survey on object detection in optical remote sensing images. *ISPRS J. Photogramm. Remote Sens.* **2016**, *117*, 11–28. [CrossRef]
4. Cheng, G.; Han, J.; Lu, X. Remote sensing image scene classification: Benchmark and state of the art. *Proc. IEEE* **2017**, *105*, 1865–1883. [CrossRef]
5. Bosch, A.; Munoz, X.; Marti, R. Which is the best way to organize/classify images by content? *Image Vis. Comput.* **2007**, *25*, 778–791. [CrossRef]
6. Michael, J.; Dana, H.; Hu, J.; Zhang, L. Color Indexing. *Int. J. Comput. Vis.* **1991**, *7*, 11–32.
7. Jain, A.; Ratha, N.; Lakshmanan, S. Object detection using gabor filters. *Pattern Recogn.* **1997**, *30*, 295–309. [CrossRef]
8. Haralick, R.; Shanmugam, K.; Dinstein, I. Textural Features for Image Classification. *Stud. Media Commun.* **1973**, *3*, 610–621. [CrossRef]

9. Ojala, T.; Pietikainen, M.; Maenpaa, T. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Trans. Pattern Anal. Mach. Intell.* **2002**, *24*, 971–987. [CrossRef]
10. Low, D. Distinctive Image Features from Scale-Invariant Keypoints. *Int. J. Comput. Vis.* **2004**, *60*, 91–110. [CrossRef]
11. Dalal, N.; Trigg, B. Histograms of Oriented Gradients for Human Detection. In Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), San Diego, CA, USA, 20–25 June 2005; pp. 886–893.
12. Oliva, A.; Torralba, A. Modeling the Shape of the Scene: A Holistic Representation of the Spatial Envelope. *Int. J. Comput. Vis.* **2001**, *42*, 145–175. [CrossRef]
13. Yang, Y.; Newsam, S. Bag-of-visual-words and spatial extensions for land-use classification. In Proceedings of the 18th SIGSPATIAL International Conference on Advances in Geographic Information Systems, San Jose, CA, USA, 2–5 November 2010; pp. 270–279.
14. Jégou, H.; Douze, M.; Schmid, C. Aggregating local descriptors into a compact image representation. In Proceedings of the 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, San Francisco, CA, USA, 13–18 June 2010; pp. 3304–3311.
15. Perronnin, F.; Dance, C. Fisher kernels on visual vocabularies for image categorization. In Proceedings of the 2007 IEEE Conference on Computer Vision and Pattern Recognition, Minneapolis, MN, USA, 17–22 June 2007; pp. 1–8.
16. Bazi, Y.; Alrahhal, M.; Alhichri, H.; Alajlan, N. Simple Yet Effective Fine-Tuning of Deep CNNs Using an Auxiliary Classification Loss for Remote Sensing Scene Classification. *Remote Sens.* **1973**, *11*, 2908. [CrossRef]
17. Long, Y. On Creating Benchmark Dataset for Aerial Image Interpretation: Reviews, Guidances, and Million-AID. *IEEE J. Sel. Top. Appl. Earth Observ. Remote Sens.* **2021**, *14*, 4205–4230. [CrossRef]
18. Chen, J.; Chen, J. Globe Land30:Operational global land cover mapping and big-data analysis. *Sci. China Earth Sci.* **2018**, *48*, 1391–1392.
19. Deng, J.; Dong, W.; Socher, R.; Li, L.; Li, K.; Li, F. ImageNet: A large-scale hierarchical image database. In Proceedings of the 2009 IEEE Conference on Computer Vision and Pattern Recognition, Miami, FL, USA, 20–25 June 2009; pp. 248–255.
20. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. ImageNet classification with deep convolutional neural networks. *Commun. ACM.* **2017**, *60*, 84–90. [CrossRef]
21. Simonyan, K.; Zisserman, A. Very Deep Convolutional Networks for Large-Scale Image Recognition. Available online: http://arxiv.org/abs/1409.1556 (accessed on 4 September 2014).
22. Szegedy, C.; Liu, W.; Jia, Y.; Sermanet P.; Rabinovich A. Going deeper with convolutions. In Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, USA, 7–12 June 2015; pp. 1–9.
23. 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.
24. Jia, Y.; Shelhamer, E.; Donahue, J.; Karayev, S; Long J.; Girshick, R.; Guadarrama, S.; Darrell, T. Caffe: Convolutional Architecture for Fast Feature Embedding. In Proceedings of the 22nd ACM International Conference on Multimedia, Orlando, FL, USA, 3–7 November 2014; pp. 675–678.
25. Jaiswal, A.; Babu, A.; Zadeh M.; Banerjee D.; Makedon F. A Survey on Contrastive Self-Supervised Learning. Available online: https://arxiv.org/abs/2011.00362 (accessed on 31 October 2020).
26. Stojni, V.; Risojevi, V. Self-Supervised Learning of Remote Sensing Scene Representations Using Contrastive Multiview Coding. Available online: https://arxiv.org/abs/2104.07070v1 (accessed on 14 Apr 2021).
27. Castelluccio, M.; Poggi, G.; Sansone, C. Land Use Classification in Remote Sensing Images by Convolutional Neural Networks. *Acta Ecol. Sin.* **2015**, *28*, 627–635.
28. Nogueira, K.; Penatti, O.; Santos, J. Towards Better Exploiting Convolutional Neural Networks for Remote Sensing Scene Classification. *Pattern Recognit.* **2016**, *61*, 539–556. [CrossRef]
29. Chaib, S.; Liu, H.; Gu, Y; Yao, H. Deep Feature Fusion for VHR Remote Sensing Scene Classification. *Pattern Recognit.* **2017**, *55*, 4775–4784. [CrossRef]
30. Penatti, O.A.B.; Nogueira, K.; dos Santos, J.A. Do deep features generalize from everyday objects to remote sensing and aerial scenes domains? In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, Boston, MA, USA, 23–28 June 2014; pp. 44–51.
31. Cheng, G.; Li, Z.; Yao, X; Guo, L; Wei Z. Remote Sensing Image Scene Classification Using Bag of Convolutional Features. *IEEE Geosci. Remote Sens. Lett.* **2017**, *14*, 1735–1739. [CrossRef]
32. Gong, X.; Liu, Y.; Xie Z. An Improved Bag-of-Visual-Word Based Classification Method for High-Resolution Remote Sensing Scene? In Proceedings of the 2018 26th International Conference on Geoinformatics, Kunming, China, 28–30 June 2018.
33. Sun, Q.; Liu, Y.; Chua, T.-S. Meta-transfer learning for few-shot learning. In Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 15–20 June 2019; pp. 403–412.
34. Wang, Y.; Yao Q.; Kwok J.T.; Ni L.M. Generalizing from a few examples: A survey on few-shot learning. *ACM Comput. Surv.* **2020**, *53*, 1–34. [CrossRef]
35. Zhang, X.; Qiang, Y.; Sung, F.; Yang Y.; Hospedales T. RelationNet2: Deep comparison network for few-shot learning. In Proceedings of the 2020 International Joint Conference on Neural Networks (IJCNN), Glasgow, UK, 19–24 July 2020; pp. 1–8.
36. Finn, C.; Abbeel, P.; Levine, S. Model-agnostic meta-learning for fast adaptation of deep networks. In Proceedings of the 34th International Conference on Machine Learning, Sydney, Australia, 6–11 August 2017; pp. 1126–1135.

37. Srinivasan, A.; Bharadwaj, A.; Sathyan, M.; Natarajan, S. Optimization of Image Embeddings for Few Shot Learning. Available online: https://arxiv.org/abs/2004.02034 (accessed on 4 April 2020).
38. Li, Z.; Zhou, F.; Chen, F.; Li, H. Meta-SGD: Learning to Learn Quickly for Few-Shot Learning. Available online: https://arxiv.org/abs/1707.09835 (accessed on 31 July 2017).
39. Zhai, M.; Liu, H.; Sun, F. Lifelong learning for scene recognition in remote sensing images. *IEEE Geosci. Remote Sens. Lett.* **2019**, *16*, 1472–1476. [CrossRef]
40. Li, L.; Han, J.; Yao, X.; Cheng, G.; Guo L. DLA-MatchNet for few-shot remote sensing image scene classification. *IEEE Trans. Geosci. Remote Sens.* **2020**, *59*, 1–10. [CrossRef]
41. Li, H.; Cui, Z.; Zhu, Z.; Chen, L.; Tao, C. Deep meta metric learning for few-shot remote sensing scene classification. *IEEE Trans. Geosci. Remote Sens.* **2020**, *58*, 1–12.
42. Li, X.; Shi, D.; Diao, X.; Xu, H. SCL-MLNet: Boosting Few-Shot Remote Sensing Scene Classification via Self-Supervised Contrastive Learning. *IEEE Trans. Geosci. Remote Sens.* **2022**, *60*, 1–12. [CrossRef]
43. Tao, C.; Qi, J.; Lu, W; Wang, H.; Li, H. Remote Sensing Image Scene Classification With Self-Supervised Paradigm Under Limited Labeled Samples. *IEEE Geosci. Remote Sens. Lett.* **2020**, *99*, 8004005. [CrossRef]
44. Zhao, Z.; Luo, Z.; Li, J; Chen, C;Piao Y. When Self-Supervised Learning Meets Scene Classification: Remote Sensing Scene Classification Based on a Multitask Learning Framework. *Remote Sens.* **2020**, *12*, 3276. [CrossRef]
45. Kang, J.; Fernandez-Beltran, R.; Duan, P; Liu, S; Plaza A.J. Deep Unsupervised Embedding for Remotely Sensed Images Based on Spatially Augmented Momentum Contrast. *IEEE Trans. Geosci. Remote Sens.* **2021**, *59*, 2598–2610. [CrossRef]
46. Rumelhart, D.E. Learning internal representations by error propagation. *Parallel Distributed Processing* **1986**, *1*, 318–363.
47. He, K.; Chen, X.; Xie, S.; Li, Y.; Dollár, P.; Girshick, R. Masked Autoencoders Are Scalable Vision Learners. Available online: https://arxiv.org/abs/2111.06377v3 (accessed on 11 November 2021).
48. Vaswani, A.; Shazeer, N.; Parmar, N; Uszkoreit, J.; Jones, L. Attention is all you need. *Adv. Neural Inf. Process. Syst.* **2017**, *30*, 5998–6008.
49. Dosovitskiy, A.; Beyer, L.; Kolesnikov, A. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. Available online: https://arxiv.org/abs/2010.11929 (accessed on 22 October 2020).
50. Xia, G.; Yang, W; Delon, J.; Gousseau, Y.; Maitre H.; Sun H. Structural high-resolution satellite image indexing. In Proceedings of the ISPRS Technical Commission VII Symposium—100 Years ISPRS—Advancing Remote Sensing Science, Vienna, Austria, 5–7 July 2010; pp. 298–303.
51. Zhu, Q.; Zhong, Y.; Zhao, B.; Xia, G.; Zhang L. Bag-of-Visual-Words Scene Classifier with Local and Global Features for High Spatial Resolution Remote Sensing Imagery. *IEEE Geosci. Remote Sens. Lett.* **2016**, *13*, 747–751. [CrossRef]
52. Li, H.; Dou, X.; Tao, C.; Wu, Z. RSI-CB: A Large Scale Remote Sensing Image Classification Benchmark via Crowdsource Data. *Sensors* **2020**, *20*, 1594. [CrossRef] [PubMed]
53. Zou, Q.; Ni, L.; Zhang, T.; Wang, Q. Deep learning based feature selection for remote sensing scene classification. *IEEE Geosci. Remote Sens. Lett.* **2015**, *12*, 2321–2325. [CrossRef]
54. Zhao, L.; Ping, T.; Huo, L. Feature significance-based multibag-of-visual-words model for remote sensing image scene classification. *J. Appl. Remote Sens.* **2016**, *10*, 035004. [CrossRef]
55. Wang, Q.; Liu, S.; Chanussot, J.; Li, X. Scene Classification With Recurrent Attention of VHR Remote Sensing Images. *IEEE Trans. Geosci. Remote Sens.* **2019**, *57*, 1155–1167. [CrossRef]
56. Zhou, W.; Newsam, S.; Li, C.; Shao Z. PatternNet: A benchmark dataset for performance evaluation of remote sensing image retrieval. *ISPRS J. Photogramm. Remote Sens.* **2018**, *145*, 197–209. [CrossRef]
57. Tong, X.; Xia, G.; Lu, Q.; Shen H. Land-cover classification with high-resolution remote sensing images using transferable deep models. *Remote Sens. Environ.* **2020**, *237*, 111322. [CrossRef]
58. Xia, G.; Hu, J.; Hu, F. AID: A benchmark data set for performance evaluation of aerial scene classification. *IEEE Trans. Geosci. Remote Sens.* **2017**, *55*, 3965–3981. [CrossRef]
59. Pour, A.M.; Seyedarabi, H.; Jahromi, S.H.A.; Javadzadeh A. Automatic detection and monitoring of diabetic retinopathy using efficient convolutional neural networks and contrast limited adaptive histogram equalization. *IEEE Access* **2019**, *8*, 136668–136673. [CrossRef]
60. Cheng, G.; Yang, C.; Yao, X.; Guo L.; Han J. When deep learning meets metric learning: Remote sensing image scene classification via learning discriminative CNNs. *IEEE Trans. Geosci. Remote Sens.* **2018**, *56*, 2811–2821. [CrossRef]
61. Liu, X.; Zhou, Y.; Zhao, J.; Yao R.; Liu B.; Zheng Y. Siamese convolutional neural networks for remote sensing scene classification. *IEEE Geosci. Remote Sens. Lett.* **2019**, *16*, 1200–1204. [CrossRef]
62. Chen, X.; Zhu, G.; Liu, M.; Wang, Q. Bag-of-Visual-Words Scene Classifier for Remote Sensing Image Based on Region Covariance. *IEEE Geosci. Remote Sens. Lett.* **2022**, *19*, 1–5. [CrossRef]
63. Tang, X.; Li, M.; Ma, J.; Zhang, X; Liu, F.; Jiao, L. EMTCAL: Efficient Multiscale Transformer and Cross-Level Attention Learning for Remote Sensing Scene Classification. *IEEE Trans. Geosci. Remote Sens.* **2022**, *60*, 1–15. [CrossRef]
64. Wang, W.; Chen, Y.; Ghamisi, P. Transferring CNN With Adaptive Learning for Remote Sensing Scene Classification. *IEEE Trans. Geosci. Remote Sens.* **2022**, *60*, 1–18. [CrossRef]
65. Zhao, Z.; Li, J.; Luo, Z.; Li, J.; Chen, C. Remote Sensing Image Scene Classification Based on an Enhanced Attention Module. *IEEE Geosci. Remote Sens. Lett.* **2021**, *18*, 1926–1930. [CrossRef]

66. Minetto, R.; Segundo, M.P.; Sarkar, S. Hydra: An ensemble of convolutional neural networks for geospatial land classification. *IEEE Trans. Geosci. Remote Sens.* **2019**, *57*, 6530–6541. [CrossRef]
67. Zhu, Q.; Zhong, Y.; Zhang, L.; Li D. Adaptive deep sparse semantic modeling framework for high spatial resolution image scene classification. *IEEE Trans. Geosci. Remote Sens.* **2018**, *56*, 6180–6195. [CrossRef]
68. Xie, J.; He, N.; Fang, L.; Plaza, A. Scale-free convolutional neural network for remote sensing scene classification. *IEEE Trans. Geosci. Remote Sens.* **2019**, *57*, 6916–6928. [CrossRef]
69. Dede, M.A.; Aptoula, E.; Genc, Y. Deep network ensembles for aerial scene classification. *IEEE Geosci. Remote Sens. Lett.* **2019**, *16*, 732–735. [CrossRef]
70. Liu,Y.; Liu,Y.; Ding, L. Scene classification by coupling convolutional neural networks with Wasserstein distance. *IEEE Geosci. Remote Sens. Lett.* **2019**, *16*, 722–726. [CrossRef]
71. Aral, R.A.; Keskin, S.R.; Kaya, M.; Haciomeroglu M. Classification of TrashNet dataset based on deep learning models. In Proceedings of the 2018 IEEE International Conference on Big Data (Big Data), Seattle, WA, USA, 10–13 December 2018; pp. 2058–2062.
72. Li, W. Classification of high-spatial-resolution remote sensing scenes method using transfer learning and deep convolutional neural network. *IEEE J. Sel. Top. Appl. Earth Observ. Remote Sens.* **2020**, *13*, 1986–1995. [CrossRef]
73. Liu, Y.; Suen, C.Y.; Liu, Y.; Ding L. Scene classification using hierarchical Wasserstein CNN. *IEEE Trans. Geosci. Remote Sens.* **2019**, *57*, 2494–2509. [CrossRef]
74. Fang, J.; Yuan, Y.; Lu, X.; Feng, Y. Robust space–frequency joint representation for remote sensing image scene classification. *IEEE Trans. Geosci. Remote Sens.* **2019**, *57*, 7492–7502. [CrossRef]
75. Sun, H.; Li, S.; Zheng, X.; Lu, X. Remote sensing scene classification by gated bidirectional network. *IEEE Trans. Geosci. Remote Sens.* **2020**, *58*, 82–96. [CrossRef]
76. Zhang, W.; Tang, P.; Zhao, L. Remote sensing image scene classification using CNN-CapsNet. *Remote Sens.* **2019**, *11*, 494. [CrossRef]