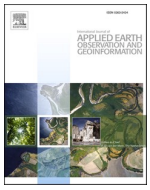
International Journal of Applied Earth Observations and Geoinformation 106 (2022) 102651

Contents lists available at ScienceDirect

International Journal of Applied Earth Observations and Geoinformation

journal homepage: www.elsevier.com/locate/jag



SITS-Former: A pre-trained spatio-spectral-temporal representation model for Sentinel-2 time series classification

Yuan Yuan [a,b,1], Lei Lin [c,1], Qingshan Liu [b,*], Renlong Hang [b], Zeng-Guang Zhou [d]

[a] School of Geographic and Biologic Information, Nanjing University of Posts and Telecommunications, Nanjing 210023, China
[b] School of Computer and Software, Nanjing University of Information Science and Technology, Nanjing 210044, China
[c] Xiaomi AI Lab, Xiaomi Inc., Beijing 100085, China
[d] Key Laboratory of Quantitative Remote Sensing Information Technology, Aerospace Information Research Institute, Chinese Academy of Sciences, Beijing 100094, China

ARTICLE INFO

Keywords:
Pre-training
satellite image time series (SITS)
Self-supervised learning
Sentinel-2
Transformer

ABSTRACT

Sentinel-2 images provide a rich source of information for a variety of land cover, vegetation, and environmental monitoring applications due to their high spectral, spatial, and temporal resolutions. Recently, deep learning-based classification of Sentinel-2 time series becomes a popular solution to vegetation classification and land cover mapping, but it often demands a large number of manually annotated labels. Improving classification performance with limited labeled data is still a challenge in many real-world remote sensing applications. To address label scarcity, we present SITS-Former (SITS stands for Satellite Image Time Series and Former stands for Transformer), a pre-trained representation model for Sentinel-2 time series classification. SITS-Former adopts a Transformer encoder as the backbone and takes time series of image patches as input to learn spatio-spectral-temporal features. According to the principles of self-supervised learning, we pre-train SITS-Former on massive unlabeled Sentinel-2 time series via a missing-data imputation proxy task. Given an incomplete time series with some patches being masked randomly, the network is asked to regress the central pixels of these masked patches based on the residual ones. By doing so, the network can capture high-level spatial and temporal dependencies from the data to learn discriminative features. After pre-training, the network can adapt the learned features to a target classification task through fine-tuning. As far as we know, this is the first study that exploits self-supervised learning for patch-based representation learning and classification of SITS. We quantitatively evaluate the quality of the learned features by transferring them on two crop classification tasks, showing that SITS-Former outperforms state-of-the-art approaches and yields a significant improvement (2.64%~3.30% in overall accuracy) over the purely supervised model. The proposed model provides an effective tool for SITS-related applications as it greatly reduces the burden of manual labeling. The source code will be released at https://github.com/linlei1214/SITS-Former upon publication.

1. Introduction

With the successive launches of the twin satellites from the Sentinel-2 mission, it is feasible to access optical Earth Observation (EO) data at much higher temporal and finer spatial resolutions than in the past. The Multi-Spectral Instrument (MSI) onboard each satellite provides 13 spectral bands with spatial resolutions ranging from 10 m to 60 m. The rapid revisit interval of 5 days with the satellite constellation increases the opportunity of acquiring cloud-free images over a geographical area, making it possible to build dense Satellite Image Time Series (SITS) from a single sensor. Owing to these improvements, Sentinel-2 time series can capture rapid variations in plant phenology with much better detail, opening the door for a variety of applications related to land cover, vegetation, and environmental monitoring (Misra et al., 2020; Phiri et al., 2020; Solano-Correa et al., 2020).

The interest in Sentinel-2 time series for vegetation and land cover mapping has gained increasing popularity in recent years. Some early studies extended methods for mono-temporal classification by extracting features from each image and then applying classical machine learning classifiers (e.g., Support Vector Machine (SVM) and Random

* Corresponding author.
E-mail address: qsliu@nuist.edu.cn (Q. Liu).
[1] Y. Yuan and L. Lin contributed equally to this work as first authors.

https://doi.org/10.1016/j.jag.2021.102651
Received 12 August 2021; Received in revised form 23 November 2021; Accepted 10 December 2021
Available online 17 December 2021

Forest (RF)) to multi-temporal features (Eudes Gbodjo et al., 2020; Feng et al., 2019; Sheeren et al., 2016). However, these methods suffer from two drawbacks. First, they consider SITS as a collection of independent images while ignoring temporal correlations between images (Interdonato et al., 2019). Second, the concatenation of multi-temporal features generally leads to high dimensionality, which results in the "curse of dimensionality" problem (Pelletier et al., 2017). Other early studies incorporated phenological information extracted from the spectral trajectory of each pixel, such as spectral statistics (Lebourgeois et al., 2017), phenological parameters (do Nascimento Bendini et al., 2019; Shahrabi et al., 2020), curve fitting coefficients (Zhu and Woodcock, 2014), *etc*. However, these hand-crafted features provide limited additional discriminative information, and their robustness is heavily affected by the selected curve fitting functions and time-series smoothing techniques.

Over recent years, deep learning approaches have achieved much success for SITS classification. Unlike traditional methods, deep learning approaches can learn discriminative features automatically without manual feature selection. Among these methods, Recurrent Neural Networks (RNNs, e.g., Long Short-Term Memory (LSTM), Gated Recurrent Unit (GRU)) (Ienco et al., 2017; Sun et al., 2019; Yuan et al., 2020), Convolutional Neural Networks (Pelletier et al., 2019; Zhong et al., 2019), and Transformers (or self-attention networks) (Rußwurm and Körner, 2020; Yuan and Lin, 2021) have been employed for pixel-based SITS classification. Several recent studies also suggest developing patch-based classification methods for Sentinel-2 time series (Benedetti et al., 2018; Garnot et al., 2019; Li et al., 2020b; Rußwurm and Körner, 2018), owing to the importance of spatial information in identifying certain land-cover/plant types. For example, fruit trees are typically regularly spaced. Hence, image texture helps to distinguish orchards from natural forests. Despite the efforts, making full use of the abundant information provided by Sentinel-2 time series is still a challenge in remote sensing.

The recent success in deep learning-based SITS classification comes at the expense of increasing the complexity of network architectures. Both the number of network layers and neurons per layer in state-of-the-art models are increasing. Training such models usually relies on millions of labeled samples, which are almost always unavailable in remote sensing applications. When labeled data is scarce, deep neural networks are very prone to overfitting, which tend to memorize the data, including the inevitable bias and noise in the training set, rather than generalize the principles underlying the data. For this reason, an overfitted network that performs perfectly on the training set performs poorly on unseen data. Although some useful regularization strategies (such as dropout (Srivastava et al., 2014), data augmentation (Perez and Wang, 2017), and normalization (Ioffe and Szegedy, 2015; Lei Ba et al., 2016)) can reduce overfitting in deep learning models, they still cannot solve this problem from the root cause.

Self-supervised learning has drawn lots of attention in the past few years due to its encouraging performance in representation learning with fewer labels (Liu et al., 2021). The main idea of self-supervised learning is to make neural networks extract general-purpose features from unlabeled data by designing proxy tasks for networks to solve. These proxy tasks are formulated by leveraging the inherent structure of the data as supervision (Jing and Tian, 2020; Kolesnikov et al., 2019). In the past few years, self-supervised learning has achieved dramatic success in Natural Language Processing (NLP) (Qiu et al., 2020) and Computer Vision (CV) (Jing and Tian, 2020) domains. However, few studies try to exploit the notion of self-supervised learning for SITS analysis. We proposed the first pre-trained representation model called SITS-BERT (Bidirectional Encoder Representations from Transformers) for Sentinel-2 time series classification in a previous study (Yuan and Lin, 2021), which has shown encouraging performance. However, SITS-BERT was still pixel-based and did not take advantage of spatial information.

To address these issues, we present SITS-Former (Former stands for Transformer), a modification of SITS-BERT for patch-based Sentinel-2 time series classification. SITS-Former uses a simple yet powerful Transformer encoder as the backbone and takes time series of image patches as input to learn spatio-spectral-temporal features. Based on the principles of self-supervised learning, we develop a novel proxy task to pre-train SITS-Former without human annotation. Given a time series of image patches (centered at an analyzed pixel) with some patches being masked, the network is made to regress the central pixels of these masked patches based on the remaining ones. We hypothesize that the features learned in thus a way can capture subtle differences between various plant species. After pre-training, the network can easily adapt the learned features to a target classification task through fine-tuning. As far as we know, this is the first attempt at self-supervised learning in patch-based representation learning using SITS.

The main contributions of this paper are summarized as follows.

(1) We propose a hybrid deep learning architecture to learn spatio-spectral-temporal features from Sentinel-2 time series.
(2) We first introduce self-supervised learning to patch-based SITS classification to address label scarcity.
(3) We carried out extensive experiments on two large-scale datasets and demonstrated the effectiveness of our method.

2. Related work

2.1. Self-supervised learning

The general pipeline of self-supervised learning is to first pre-train a network on massive unlabeled data and then fine-tune it on a few labels related to a target task. Pre-training is the process of learning features via pre-defined proxy tasks, while fine-tuning refers to the process of updating the model parameters to solve a given task. Currently, this unsupervised-pre-training + supervised-fine-tuning pipeline has dominated NLP and shown powerful skills (Devlin et al., 2019). Self-supervised methods have also started to flourish recently in CV and achieved remarkable performance that even match that of purely supervised methods on standard benchmarks (Chen et al., 2020).

The key components of an effective self-supervised method are proxy tasks, which should allow networks to learn high-level abstract representations (Kolesnikov et al., 2019). In general, a proxy task is designed in such a way that predicts any part of the input from other parts in some form (Qiu et al., 2020). In NLP, some effective proxy tasks include language modeling, masked language modeling, next sentence prediction, sentence order prediction, *etc*. In CV, proxy tasks are more diverse and still increasing (Jing and Tian, 2020). Visual information used for self-supervision includes spatial context structures, colors, multi-mode correspondences, temporal contiguity between video frames, *etc*.

Despite these achievements, research and applications of self-supervised learning in remote sensing have just begun. Recent studies explored the use of self-supervised learning in addressing tasks such as image registration (Wang et al., 2018), change detection (Dong et al., 2020; Saha et al., 2020), scene classification (Tao et al., 2020; Zhao et al., 2020b), and image classification (Li et al., 2021; Vincenzi et al., 2021; Yue et al., 2021). However, these approaches have been restricted to analyzing a single image or bi-temporal images without considering temporal information. In contrast, our study focuses on learning spatial and temporal features from time series of satellite images, facilitating downstream tasks such as vegetation classification and land cover mapping.

2.2. SITS classification

We briefly review popular deep learning models for SITS classification, namely, CNNs, RNNs, Transformers, and their combinations.

CNN has been extensively studied in a variety of remote sensing applications, such as multispectral/hyperspectral image classification

(Hang et al., 2020; Hang et al., 2021; Zhang et al., 2018), scene classification (Anwer et al., 2018; Cheng et al., 2018), image segmentation (Waldner and Diakogiannis, 2020), object detection (Cheng et al., 2016), *etc*. Pelletier et al. (2019) introduced Temporal Convolutional Neural Networks (TempCNNs) for SITS-based crop mapping, where convolutions are operated in the time domain to capture structures of spectral profiles. Ji et al. (2018) applied 3D-CNNs on multi-temporal images to extract spatio-temporal-combined representations.

RNN is another commonly used architecture for processing sequential data. Ienco et al. (2017) introduced LSTM for SITS-based land cover classification. Rußwurm and Körner (2017) adopted a similar network for vegetation development modeling, except that the inputs were flattened local image patches. Wang et al. (2019) utilized a Bidirectional LSTM (Bi-LSTM) network to aggregate the temporal information from both past and future contexts. Several hybrid convolutional and recurrent architectures have also been developed for SITS analysis (Garnot et al., 2019; Interdonato et al., 2019; Rußwurm and Körner, 2018).

Transformer was introduced by Vaswani et al. (2017) and has since become state-of-the-art in NLP. In contrast to RNNs that operate sequence elements one at a time recursively, Transformers operate on all elements in parallel and model the influence each element has on another by assigning attention scores (Minaee et al., 2021). Transformers overcome the difficulty of RNNs for remembering long-term temporal dependencies and allow for more parallelization. Rußwurm and Körner (2020) carried out comprehensive experiments to evaluate the performance of several deep learning models for SITS classification, showing that Transformers and RNNs outperform TempCNNs in handling noisy raw Sentinel-2 time series. Yuan and Lin (2021) found that Transformers are more robust to overfitting than RNNs and TempCNNs with limited training samples. Li et al. (2020b) developed a CNN-Transformer hybrid architecture for crop classification.

In summary, the existing methods extract features either in the time domain or in the spatio-temporal domains. In contrast, SITS-Former is designed to capture multi-dimensional dependencies from the data in spectral, spatial, and temporal domains simultaneously.

3. Methods

In this study, we use all spectral bands at 10 ∼ 20 m resolutions of Sentinel-2 images. The images are pre-processed to Bottom-Of-Atmosphere (BOA) reflectance with an equal resolution for all bands. Fig. 1 illustrates the construction of an image patch time series. Only noise-free patches are used. If a patch contains noisy pixels such as clouds, snow, or shadows according to the scene classification map, the patch is eliminated from a time series. Let $x = \{x_1, \cdots, x_N\}$ be an annual time series of N timesteps, where each element is a tuple $x_i = \langle O_i, t_i \rangle$. O_i represents the ith local image patch centered at an analyzed pixel (the pixel to be classified). The size of the patches is 5×5. Therefore, O_i is a 3D tensor of size $(10 \times 5 \times 5)$. t_i represents the acquisition time of O_i, which is expressed as the Julian date (i.e., Day Of Year (DOY)). The length of time series can be different due to irregular sampling.

3.1. Architecture of SITS-Former

SITS-Former comprises two modules, i.e., an image patch embedding module and a Transformer encoder module (Fig. 2). The image patch embedding module operates on all the patches independently to learn inner-patch spatio-spectral features; it transforms an input time series into a set of corresponding embedding vectors $\{E_1, \cdots, E_N\}$. The Transformer encoder module enables interactions between the patches to learn temporal relations amongst different timesteps; it takes the embedding vectors as input and transforms them into their temporal-context-enriched representations $\{T_1, \cdots, T_N\}$.

Image patch embedding module: The image patch embedding module aims to transform each input tuple $\langle O_i, t_i \rangle$ into a single embedding vector E_i, which is the sum of two vectors of the same dimension: a learnable spatio-spectral embedding vector (SSEV) denoted by $SSEV(O_i)$, and a pre-defined positional encoding vector (PEV) denoted by $PEV(t_i)$.

SSEV is produced by processing each input patch with a lightweight network (Fig. 3). The network is composed of two 3-Dimensional Convolutional (Conv3D) layers (32 filters with $5 \times 3 \times 3$ kernel size in the first layer, and 64 filters with $3 \times 3 \times 3$ kernel size in the second layer), a flatting layer, and a Fully Connected (FC) layer with 256 hidden nodes. Each Conv3D layer is followed by a Rectified Linear Unit (ReLU) activation function to inject nonlinearity and a Batch Normalization (BN) layer to enable faster and more stable training (Ioffe and Szegedy, 2015). Compared to CNN-Transformer (Li et al., 2020b), we use 3D-CNNs instead of traditional 2D-CNNs to exploit inter-band correlation and local image structures simultaneously.

PEV encodes the acquisition time of an image, allowing the network to cope with irregularly sampled time series and align dates from different years. In general, PEV can be either fixed or learnable. In this study, we adopt the sinusoidal positional encoding technique (Vaswani et al., 2017) to calculate PEV. A pre-defined vector is assigned to each DOY and no extra parameter is introduced.

$$PE(t_i)_p = \begin{cases} \sin\left(\frac{t_i}{10000^{\frac{k}{D}}}\right), \text{if} p = 2k \\ \cos\left(\frac{t_i}{10000^{\frac{k}{D}}}\right), \text{if} p = 2k+1 \end{cases} \quad (1)$$

where $PE(t_i)_p$ represents the pth element of PEV; k is an auxiliary variable indicating the parity of p.

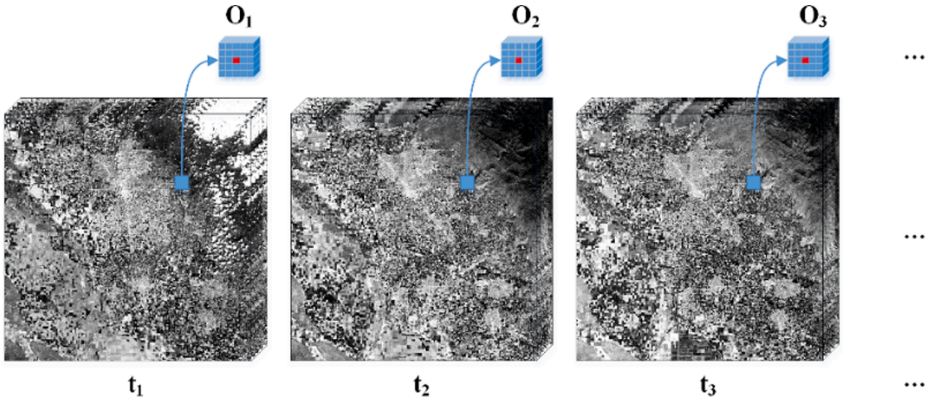


Fig. 1. Illustration of an image patch time series $\{\langle O_i, t_i \rangle\}$, where O_i is the ith image patch centered at an analyzed pixel (colored in red) and t_i is the acquisition time. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

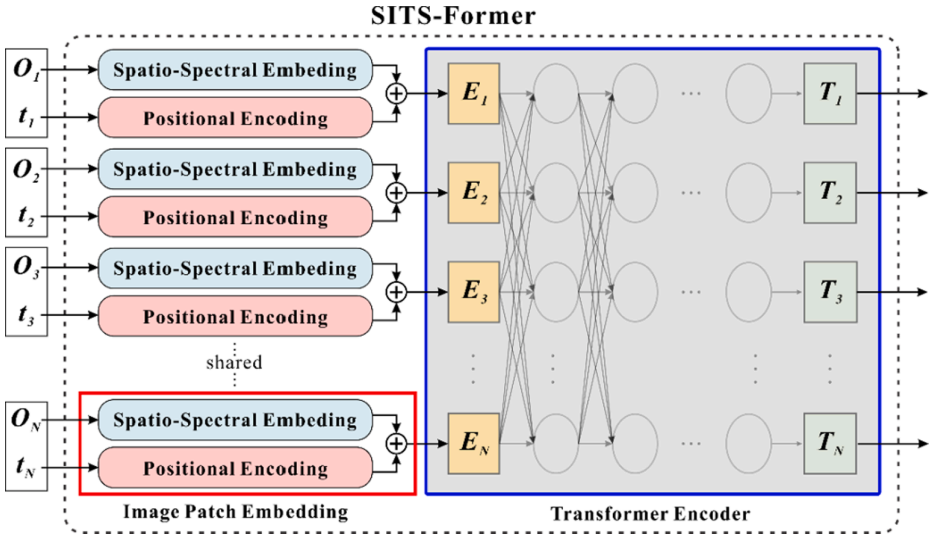


Fig. 2. SITS-Former is composed of two modules: an image patch embedding module (marked with a red rectangle) and a Transformer encoder module (marked with a blue rectangle). The image patch embedding module is shared across different timesteps. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

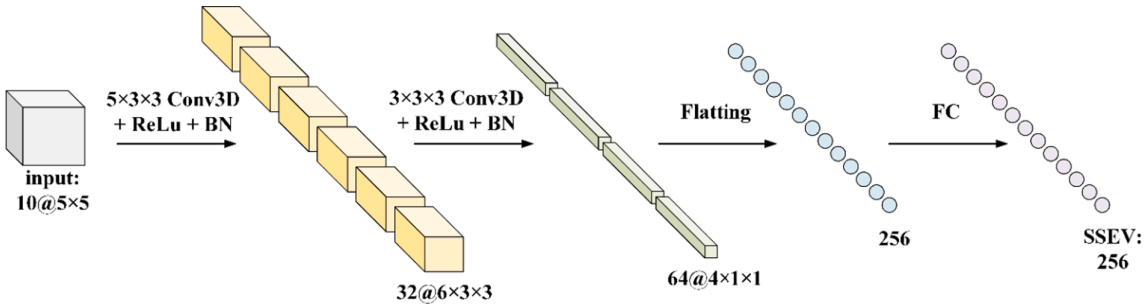


Fig. 3. Illustration of the spatio-spectral embedding network.

Transformer encoder module: We adopt a multi-layer Transformer encoder to integrate the temporal contextual information into the patches' representations. The basic building blocks of a Transformer encoder are multi-head self-attention layers. Each layer is composed of multiple identical sub-layers called heads. In each head, an attention matrix is calculated by the scaled dot-product of a query matrix (Q) and a key matrix (V) multiplying with a value matrix (V). The Q, K, and V matrices of each head are obtained by processing the sequence elements with different FC layers. The mathematical description of the multi-head attention mechanism is:

$$\text{Attention}(Q,K,V) = \text{softmax}\left(\frac{QK^T}{\sqrt{D}}\right)V \tag{2}$$

$$\text{MultiHead}(Q,K,V) = \text{Concat}(head_1,\cdots,head_H)$$

$$\text{where} head_i = \text{Attention}\left(QW_i^Q, KW_i^K, VW_i^V\right). \tag{3}$$

where H is the number of heads; W_i^Q, W_i^K, W_i^V are learnable weight matrices belonging to each head.

3.2. Pre-training of SITS-Former

In this section, we propose a novel framework for applying SITS-Former to an interested classification task. The proposed framework includes two stages: a self-supervised pre-training stage and a supervised fine-tuning (or classification) stage (Fig. 4). Apart from an additional output module, SITS-Former is used in both stages as a shared feature extractor, which allows the well-learned features during pre-training to be transferred to a target task.

Self-supervised pre-training stage: Our objective is to learn discriminative features from unlabeled data without resorting to any human annotation. We design a novel proxy task that trains the network to complete the missing part of an input time series. We refer to this task as "missing-data imputation". Suppose we are given a set of unlabeled annual time series of image patches. For each time series, a certain proportion of patches are randomly replaced with a special padding matrix [MASK] before feeding it to the network. The padding matrix is of the same dimension as the original patches and is shared between all masked timesteps. The elements of [MASK] are random numbers from a normal distribution. The training objective requires recovering the analyzed pixels at all the masked timesteps conditioned on their acquisition dates (Fig. 4). By this means, the network has to learn the spatiotemporal contextual relations between the patches to fill in the missing content. In our implementation, 15% of patches in each time series are masked, which are selected randomly for each training epoch to increase the difficulty of this task.

Formally, given an unlabeled pre-training set $\{x^{(1)},\cdots,x^{(m)}\}$ of m samples. Let $v^{(s)} = \left[v_1^{(s)},\cdots,v_N^{(s)}\right]$ be the reflectance time series associated with the analyzed pixel of sample $x^{(s)}$. We use a single FC layer to map

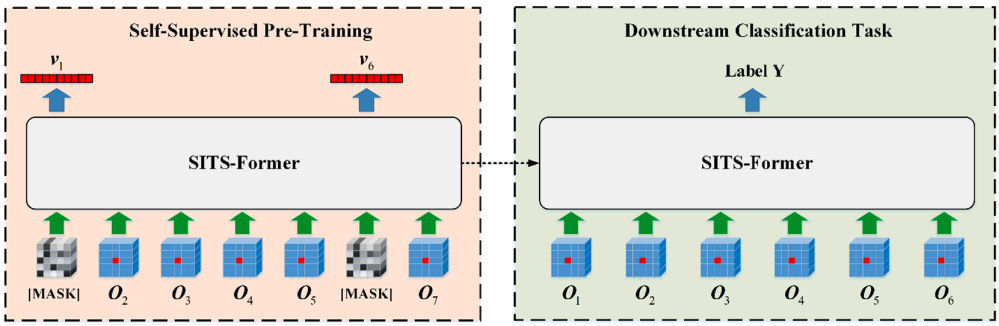


Fig. 4. The proposed framework consists of a self-supervised pre-training stage and a supervised fine-tuning (classification) stage. In the pre-training stage, general features are learned from unlabeled Sentinel-2 time series by training SITS-Former to solve a pre-defined proxy task. Afterward, in the fine-tuning stage, the learned parameters can be further transferred to a specific SITS classification task as a pre-trained model to improve performance and prevent overfitting.

the outputs of SITS-Former (i.e., $T^{(s)} = \left[T_1^{(s)}, \cdots, T_N^{(s)}\right]$) into predictions, and use the L2 distance between the predictions and targets at all masked positions as the loss function:

$$\mathscr{L} = \sum_{s=1}^{m} \mathrm{M}^{(s)} \odot \left(\mathrm{v}^{(s)} - \mathrm{W_p T}^{(s)}\right)_2. \tag{4}$$

where W_p is a learnable weight matrix; $M^{(s)}$ is a binary vector corresponding to $x^{(s)}$, with its element of 1 indicating a timestep being masked and 0 indicating non-masked; $\odot$ represents the element-wise product; $\|\bullet\|_2$ represents l_2-norm.

The basic idea behind this task is to make neural networks emulate human inductive reasoning. For example, for a crop classification task, experienced human analysts only need a few images taken during the growing season to identify crop types, benefiting from their prior knowledge about plant appearance and growth patterns. Similarly, if a network can discover meaningful spatiotemporal patterns related to the processes of vegetation growth and seasonality from massive unlabeled data, it is possible to match an unseen time series with these patterns even if the vegetation type is not yet known. Moreover, the knowledge accumulated through pre-training may aid the acquisition of new knowledge, such as plants that do not exist in the pre-training set, because some morphological (like plant spacing and ridge spacing), phenological (such as sowing, emergence, tasseling, maturation, and harvest), and spectral characteristics are shared among various species. Note that we neither regress the entire time series nor reconstruct the whole masked patches. On one hand, time series regression is a much easier task than missing-data imputation since most sequence elements are known. On the other hand, we want the network to focus on an analyzed pixel instead of trivial texture statistics.

Data augmentation is an effective technique to help networks learn invariant features and avoid overfitting. We adopt three data augmentation strategies to generate more training samples: scale, flip, and rotation. Specifically, scale indicates resampling patches to 10 m or 20 m while keeping the size of the patches unchanged (the corresponding area on the Earth's surface is not equal). Flip indicates flipping images horizontally or vertically. Rotation means that images are rotated clockwise by 0°, 90°, 180°, or 270°. The same augmentation technique is applied to all patches in a time series.

Supervised fine-tuning stage: After pre-training, an additional classification module, which is randomly initialized, is added on top of SITS-Former. We adopt a simple classification module, which is formed by a max-pooling layer and a single FC layer. The outputs of SITS-Former at all timesteps are first pooled along the temporal dimension and then linearly projected for label prediction. When dealing with a specific task, we directly load the pre-trained weights of SITS-Former and then fine-tune the entire model end-to-end on task-related data via standard supervised learning.

4. Research data

4.1. Study areas

A common practice in self-supervised learning is to verify the quality of the learned features by solving a downstream task. Here, we concentrated our analysis on crop classification in two study sites (each 100 × 100 km^2, Fig. 5). The first study area was located in the City of Fresno, California, from Sentinel-2 tile T11SKA. The elevation ranges from 40 m to 2000 m. This area has a Mediterranean climate with mild, wet winters and hot, dry summers. The second study site was located on the boundary between Missouri and Arkansas, from Sentinel-2 tile T15SYA. The elevation ranges from 60 m to 270 m. This area has a humid subtropical climate with cold to mild winters and hot, humid summers. The crop types in both areas are very diverse, and there are differences between the two regions in climate, topography, and plant species. Therefore, they can fully verify the effectiveness and generalization performance of our model.

4.2. Data preparation

We downloaded Sentinel-2A/B images with cloud coverage lower than 10% and pre-processed them into Level-2A products using the Sen2Cor plugin and the Sentinel Application Platform (SNAP).

To construct a pre-training set, we collected 216 images during 2018 to 2019 from three tiles: T11SKA, T10SEJ, and T10SFH (Fig. 6, Table 1), which are located in the Central Valley of California. The plant species in this region are extremely rich, making it a good training ground for self-supervised learning (Yuan and Lin, 2021). A pre-training sample was generated by the following steps. We first cropped small patches from the images over the same tile at an interval of 20 rows/columns, and then arranged the patches centered at the same pixel into time series. The patches was of size 5 × 5 at a resolution of 10 m or 20 m after resampling. There was no overlap between adjacent patches, ensuring the difference between two samples large enough. Time series with less than 10 timesteps were excluded. Finally, there were around 1.66 million unlabeled samples in the pre-training set.

For method evaluation, we collected 45 images and 26 images captured during 2019 in each study area, respectively. The 2019 Cropland Data Layer (CDL) (Boryan et al., 2011) and its corresponding confidence layer (Liu et al., 2004) were used as reference data to collect samples. CDL data have been widely used as benchmarks for crop classification due to their high quality (Li et al., 2020a; Sun et al., 2019; Zhao et al., 2020a). We resampled the Sentinel-2 images and the reference data to 10 m to make them consistent. Then we used the same sampling strategy to extract image patch time series, and labeled each time series with the crop type of the patch's central pixel. To avoid the adverse effects of noisy labels, a qualified sample should meet the following conditions. First, more than 50% of the pixels in the entire

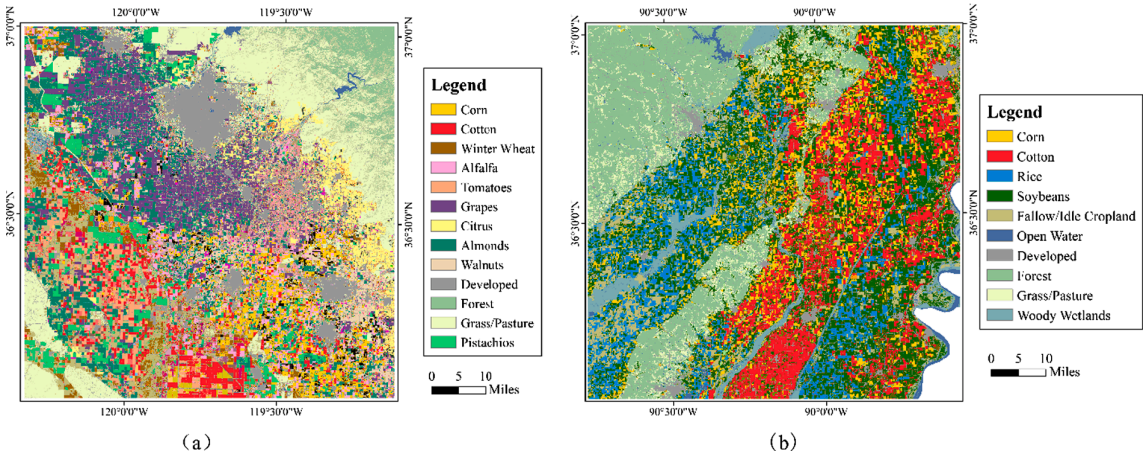


Fig. 5. Location of the two study sites: (a) California, (b) Missouri. The background images are 2019 Cropland Data Layer (CDL) maps.

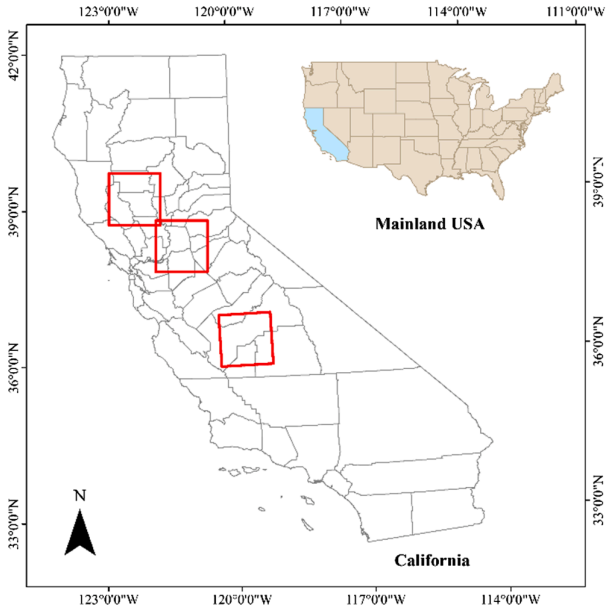


Fig. 6. Location of the Sentinel-2 tiles (marked with red rectangles) used for constructing the pre-training set. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

Table 1
The elaborate information of the Sentinel-2 images used for pre-training.

Sentinel-2 tile	Number of images (cloud cover less than 10%)	Top 5 major crops
T11SKA	41 in 2018, 45 in 2019	Almonds, Grapes, Cotton, Pistachio, Alfalfa
T10SEJ	30 in 2018, 38 in 2019	Rice, Almonds, Walnuts, Tomatoes, Winter Wheat
T10SFH	39 in 2018, 23 in 2019	Alfalfa, Grapes, Walnuts, Corn, Winter Wheat

patch should have the same label as the central pixel. Second, the classification confidences of these pixels should be higher than 80%. Finally, time series with less than 3 timesteps were discarded. Table 2 summarizes the number of samples selected from each study site.

The labeled samples were further divided into training, validation, and test sets. The training set was used for training (or fine-tuning) networks, while the validation set was used for hyper-parameter optimization. The best-performing model on the validation set was used for accuracy assessment. To simulate real situations of label scarcity, we randomly drew 100 labeled samples per class to construct the training and validation sets independently and then assigned all remaining samples to the test set. As a result, for the first study site (California), the training and validation sets each contained 1300 samples, and the test set contained 323,150 samples. For the second study site (Missouri), the training and validation sets each contained 1000 samples, and the test set contained 390,026 samples. These test sets were hard to classify because their class distributions were highly imbalanced.

Table 2
Number of samples collected from each study site.

Study area	Crop type	Number of samples	Percentage of each class (%)	Total number of samples
California	Corn	3888	1.2	324,450
	Cotton	25,014	7.7	
	Winter Wheat	2310	0.7	
	Alfalfa	14,860	4.6	
	Tomatoes	13,191	4.1	
	Grapes	66,984	20.6	
	Citrus	11,886	3.7	
	Almonds	54,462	16.8	
	Walnuts	14,578	4.5	
	Developed	1746	0.5	
	Forest	3396	1.0	
	Grass/ Pasture	77,112	23.8	
	Pistachios	35,023	10.8	
Missouri	Corn	45,264	11.6	391,026
	Cotton	83,917	21.5	
	Rice	56,589	14.5	
	Soybeans	116,656	29.8	
	Fallow	14,264	3.6	
	Open Water	5933	1.5	
	Developed	1347	0.3	
	Forest	43,051	11.0	
	Grass/ Pasture	2071	0.5	
	Woody Wetlands	21,934	5.6	

5. Results

5.1. Experimental settings and evaluation metrics

In our experiments, we compared SITS-Former with five competing methods. We employed RF as a traditional machine learning baseline owing to its superiority for dealing with high-dimensional data (Pelletier et al., 2016). Besides, we selected four advanced deep learning models that have shown promising results for SITS classification: SITS-BERT (Yuan and Lin, 2021), CNN-Transformer (Li et al., 2020b), Convolutional Recurrent Neural Network (ConvRNN) (Rußwurm and Körner, 2018), and DUal view Point deep Learning architecture for time series classificatiOn (DuPLO) (Interdonato et al., 2019). For CNN-Transformer, ConvRNN, and DuPLO, we made the input and output consistent with our model. For RF and SITS-BERT, the input was the reflectance time series of the analyzed pixel.

For RF, we selected the best tree number in range (100, 200, 300, 400, 500). For ConvRNN, we utilized a Bi-GRU network with 256 hidden states and 3×3 kernel size in each layer. For DuPLO and SITS-BERT, we adopted the same configurations as the authors suggested. For CNN-Transformer, we adopted an architecture similar to SITS-Former but replacing 3D-CNNs with 2D-CNNs. As seen from Table 3, the model size of SITS-Former is similar to SITS-BERT and CNN-Transformer, while much smaller than ConvRNN and DuPLO. Using the design choices discussed above, SITS-Former saves around 82% of DuPLO's parameters.

For SITS-Former, we first pre-trained it for 200 epochs and then fine-tuned it on task data for 30 epochs. Adam optimizer (Kingma and Ba, 2014) with a batch size of 128 was used for both pre-training and fine-tuning. In the pre-training stage, the initial learning rate was set to 1e-4. The learning rate warmed up over the first 10 epochs and then decayed by gamma every epoch. The value of gamma was set to 0.99. To avoid overfitting, dropout was applied to the output of all layers of SITS-Former with a dropout rate of 0.1. We also trained all the competing models from scratch for 200 epochs with a learning rate of 1e-4. In order to objectively evaluate the quality of the pre-trained model, we also considered the pre-trained versions of SITS-BERT (Yuan and Lin, 2021) and CNN-Transformer, where CNN-Transformer was pre-trained in the same way as SITS-Former. In all experiments, data augmentation was performed on the training sets.

We adopted three evaluation metrics to quantify the classification performance of different methods: Overall Accuracy (OA), Average Accuracy (AA), and Kappa coefficient. OA and Kappa coefficient reflect the quality of classification in terms of quantity and consistency respectively, while AA reflects the average performance in all classes. Besides, we adopted the F1-score to quantify the classification accuracy of each class (Rußwurm and Körner, 2020). All the results were averaged over five random data splits.

5.2. Ablation study

We first analyze the impact of each component of SITS-Former on the classification accuracy by comparing it with three variants: (1) removing the use of batch normalization (SITS-Former$_{\text{withoutBN}}$), (2) removing the use of positional encoding (SITS-Former$_{\text{withoutPE}}$), (3) using a BERT-style learnable positional encoding (SITS-Former$_{\text{learnablePE}}$) (Devlin et al., 2019). Tables 4-5 report the results on both datasets.

We observe that SITS-Former globally outperforms its variants without BN or PEV in terms of AA and Kappa coefficient. The AA decrease caused by removing BN is 0.51% and 1.02% on the test set for each dataset, respectively. Similarly, the AA decrease caused by removing PEV is 1.36% and 0.74%, respectively. These results confirm that both batch normalization and positional encoding play positive roles in our model. We also observe that using a learnable PEV instead of a fixed one does not bring significant improvement in accuracy. On the contrary, all evaluation metrics obtained by SITS-Former$_{\text{learnablePE}}$ on both validation sets are lower than those obtained by SITS-Former. This may be because the lengths of annual satellite time series are short, so the use of learnable positional encodings cannot significantly affect the result, but may increase the risk of overfitting. This founding is consistent with Wang and Chen (2020), who found that fixed sinusoidal positional encoding method performs perfectly on short sequences.

Table 3
Number of trainable parameters of different deep learning models for the California dataset.

Model	Number of parameters (million)
SITS-BERT	2.4
CNN-Transformer	2.5
SITS-Former	2.5
ConvRNN	10.2
DuPLO	14.1

Except for DuPLO, the number of parameters of other models are independent of the input length.

5.3. Comparison with state-of-the-art methods

The classification accuracy of all competing methods is reported in Tables 6-7. Regarding non-pre-trained models, SITS-Former achieves the best performance in most cases. Specifically, SITS-Former and CNN-Transformer perform equally well on the test sets, while the former outperforms the latter on both validation sets. In addition, the pre-trained SITS-Former performs the best among all the comparison methods on both datasets, followed by the pre-trained CNN-Transformer. Both of them yield remarkable improvements over their non-pre-trained counterparts. Specifically, for SITS-Former, the accuracy gains brought by pre-training are 3.30% OA / 3.04% AA on the California test set, and 2.64% OA / 3.29% AA on the Missouri test set. For CNN-Transformer, the accuracy gains brought by pre-training are 3.10% OA / 2.81% AA on the California test set, and 3.14% OA / 3.11% AA on the Missouri test set.

In order to verify whether pre-training significantly improves the classification performance, we performed the one-way ANalysis Of VAriance (ANOVA) test on the results of five random runs reported in Tables 6-7 to evaluate the null hypothesis, i.e., whether pre-training a model does not affect the final classification accuracy. This test produced a p-value for each evaluation metric. Metrics with p-values less than 0.05 were considered significant, meaning that the null hypothesis was rejected. As shown in Table 8, the p-values associated with all evaluations metrics on both datasets are extremely low. These results support our hypothesis that SITS classification tasks can greatly benefit from self-supervised pre-training when there is insufficient labeled data, even if the pre-training data and task data come from different data distributions, just like the case of Missouri.

We further investigate the performance of different methods on each class. The per-class F1-scores obtained by all competing methods on test sets are reported in Tables 9 and 10. The corresponding confusion matrices are visualized in Figs. 7 and 8. The confusion matrices are normalized by row, whose diagonal elements represent producer's accuracy. We observe that the pre-trained SITS-Former achieves higher F1-scores on almost all classes over its non-pre-trained counterpart. For the California dataset, the pre-trained SITS-Former achieves the highest F1-scores on 8 out of 13 classes. The highest performance gains induced by pre-training are observed on winter wheat, developed, walnuts, and pistachios with an improvement of 4.64%~11% F1-score. Similarly, for the Missouri dataset, the pre-trained SITS-Former achieves the highest F1-scores on 7 out of 10 classes. The highest performance gains are observed on developed, grass/pasture, and woody wetlands with an improvement of 5.29%~9.40% F1-score.

In general, it is difficult to distinguish between permanent crops such as grape vineyards and nut trees (such as walnuts, almonds, and

Table 4
Accuracy assessment for the California dataset considering different variants of SITS-Former.

Study area	Model	Validation set			Test set		
		OA (%)	AA (%)	Kappa	OA (%)	AA (%)	Kappa
California	SITS-Former$_{withoutBN}$	87.33 ± 0.30	87.33 ± 0.30	0.8628 ± 0.0032	85.55 ± 0.39	86.66 ± 0.18	0.8335 ± 0.0042
	SITS-Former$_{withoutPE}$	86.38 ± 0.76	86.38 ± 0.76	0.8517 ± 0.0100	84.49 ± 0.42	85.81 ± 0.27	0.8217 ± 0.0048
	SITS-Former$_{learnablePE}$	87.51 ± 0.35	87.51 ± 0.35	0.8647 ± 0.0037	**85.56** ± 0.37	**87.22** ± 0.30	**0.8339** ± 0.0041
	SITS-Former	**88.41** ± 0.80	**88.41** ± 0.80	**0.8744** ± 0.0086	85.53 ± 0.47	87.17 ± 0.42	**0.8339** ± 0.0049

Best results are highlighted in bold.
Note: On the validation set, OA is equal to AA.

Table 5
Accuracy assessment for the Missouri dataset considering different variants of SITS-Former.

Study area	Model	Validation set			Test set		
		OA (%)	AA (%)	Kappa	OA (%)	AA (%)	Kappa
Missouri	SITS-Former$_{withoutBN}$	88.23 ± 0.53	88.23 ± 0.53	0.8693 ± 0.0059	89.87 ± 0.42	86.42 ± 0.39	0.8786 ± 0.0049
	SITS-Former$_{withoutPE}$	87.27 ± 0.62	87.27 ± 0.62	0.8585 ± 0.0069	89.48 ± 0.49	86.70 ± 0.35	0.8742 ± 0.0057
	SITS-Former$_{learnablePE}$	87.13 ± 0.48	87.13 ± 0.48	0.8570 ± 0.0053	89.73 ± 0.47	87.31 ± 0.15	0.8773 ± 0.0055
	SITS-Former	**88.50** ± 0.50	**88.50** ± 0.50	**0.8722** ± 0.0055	**90.54** ± 0.40	**87.44** ± 0.16	**0.8868** ± 0.0046

Table 6
Accuracy assessment considering different competing methods on the California dataset.

Method	Validation set			Test set		
	OA (%)	AA (%)	Kappa	OA (%)	AA (%)	Kappa
RF	84.26 ± 0.45	84.26 ± 0.45	0.8294 ± 0.0049	82.91 ± 0.44	84.17 ± 0.34	0.8034 ± 0.0049
SITS-BERT	85.61 ± 0.06	85.61 ± 0.06	0.8442 ± 0.0007	83.65 ± 1.14	84.38 ± 0.17	0.8118 ± 0.0125
ConvRNN	84.31 ± 0.33	84.31 ± 0.33	0.8300 ± 0.0036	81.46 ± 0.39	82.82 ± 0.28	0.7871 ± 0.0042
DuPLO	86.51 ± 0.70	86.51 ± 0.70	0.8539 ± 0.0076	84.50 ± 0.25	86.20 ± 0.19	0.8219 ± 0.0027
CNN-Transformer	87.87 ± 0.36	87.87 ± 0.36	0.8686 ± 0.0039	**85.57** ± 0.75	87.07 ± 0.56	0.8336 ± 0.0084
SITS-Former	**88.41** ± 0.80	**88.41** ± 0.80	**0.8744** ± 0.0086	85.53 ± 0.47	**87.17** ± 0.42	**0.8339** ± 0.0049
SITS-BERT$_{pretrained}$	89.59 ± 0.50	89.59 ± 0.50	0.8872 ± 0.0055	87.88 ± 0.08	89.12 ± 0.25	0.8603 ± 0.0010
CNN-Transformer$_{pretrained}$	90.51 ± 0.04	90.51 ± 0.04	0.8972 ± 0.0004	88.67 ± 0.31	89.88 ± 0.25	0.8693 ± 0.0036
SITS-Former$_{pretrained}$	**90.69** ± 0.29	**90.69** ± 0.29	**0.8992** ± 0.0031	**88.83** ± 0.26	**90.21** ± 0.12	**0.8806** ± 0.0150

The model with subscript 'pretrained' means that the model has been pre-trained.

Table 7
Accuracy assessment considering different competing methods on the Missouri dataset.

Method	Validation set			Test set		
	OA (%)	AA (%)	Kappa	OA (%)	AA (%)	Kappa
RF	86.00 ± 0.71	86.00 ± 0.71	0.8444 ± 0.0079	87.98 ± 0.45	86.15 ± 0.10	0.8569 ± 0.0051
SITS-BERT	83.37 ± 0.56	83.37 ± 0.56	0.8152 ± 0.0062	87.38 ± 0.33	83.25 ± 0.37	0.8487 ± 0.0040
ConvRNN	85.83 ± 0.54	85.83 ± 0.54	0.8426 ± 0.0060	85.87 ± 0.14	83.88 ± 0.89	0.8316 ± 0.0018
DuPLO	87.43 ± 1.36	87.43 ± 1.36	0.8604 ± 0.0151	89.21 ± 0.28	86.69 ± 0.19	0.8711 ± 0.0032
CNN-Transformer	88.43 ± 0.21	88.43 ± 0.21	0.8715 ± 0.0023	89.88 ± 0.47	**87.59** ± 0.09	0.8793 ± 0.0054
SITS-Former	**88.50** ± 0.50	**88.50** ± 0.50	**0.8722** ± 0.0055	**90.54** ± 0.40	87.44 ± 0.16	**0.8868** ± 0.0046
SITS-BERT$_{pretrained}$	89.60 ± 0.43	89.60 ± 0.43	0.8844 ± 0.0048	92.16 ± 0.28	89.44 ± 0.35	0.9061 ± 0.0032
CNN-Transformer$_{pretrained}$	91.00 ± 0.65	91.00 ± 0.65	0.9000 ± 0.0072	93.02 ± 0.23	90.70 ± 0.10	0.9163 ± 0.0027
SITS-Former$_{pretrained}$	**91.33** ± 0.66	**91.33** ± 0.66	**0.9037** ± 0.0074	**93.18** ± 0.32	**90.73** ± 0.12	**0.9184** ± 0.0039

Table 8
Results of the one-way ANOVA significance test. Tests are conducted between the pre-trained and non-pre-trained models of SITS-Former and CNN-Transformer.

Method	California			Missouri		
	OA	AA	Kappa	OA	AA	Kappa
CNN-Transformer	5.821e-03	2.858e-03	5.426e-03	1.035e-03	4.703e-06	9.459e-04
SITS-Former	4.308e-07	6.021e-04	1.380e-02	1.883e-03	1.812e-05	1.729e-03

pistachios) due to their similar growth patterns: vegetation vigor in mid-spring and defoliation in late-summer. The spectral trajectories of these crops mainly depend on phenological shifts, which are mostly influenced by stable local climate and regional factors rather than agricultural practices (Zhong et al., 2011). We also find that many developed pixels are incorrectly classified as winter wheat and grass/pasture, resulting in low F1-scores on these classes. This is because developed is a mixed land-use type formed by merging four types of

Table 9
Per-class F1-score (%) for the California dataset considering different competing methods.

Method	Corn	Cotton	WinterWheat	Alfalfa	Tomatoes	Grapes	Citrus	Almonds	Walnuts	Developed	Forest	Grass/ Pasture	Pistachios
RF	85.77	92.49	39.49	85.22	93.35	81.10	80.08	85.77	77.80	50.14	87.71	90.04	80.54
SITS-BERT	80.19	94.08	45.89	85.31	93.50	81.68	79.92	85.65	75.01	48.77	85.22	90.78	80.33
ConvRNN	77.49	91.87	42.61	82.32	92.13	79.01	77.81	85.16	73.58	46.85	86.35	89.59	77.53
DuPLO	85.84	93.62	54.48	85.24	92.82	81.58	81.44	85.49	77.80	50.32	87.80	91.88	86.17
CNN-Transformer	88.49	94.69	52.60	85.51	94.52	82.61	84.09	86.29	77.93	53.23	**92.43**	92.22	85.43
SITS-Former	87.70	94.86	48.73	87.47	94.22	84.27	84.39	86.41	79.05	50.73	90.31	91.40	86.65
SITS-BERT$_{pretrained}$	87.13	95.43	52.03	90.21	94.55	86.51	**88.51**	88.85	81.24	57.22	89.99	92.99	90.08
CNN-Transformer$_{pretrained}$	**89.10**	96.03	**62.11**	91.02	95.58	86.58	88.16	**89.07**	82.74	59.26	90.38	93.62	89.89
SITS-Former$_{pretrained}$	87.32	**96.06**	59.73	**91.70**	**95.98**	**86.89**	87.01	88.61	**85.96**	**59.57**	90.76	**93.76**	**91.29**

developed land, i.e., open space, low, medium, and high intensity. Hence, the large misclassification may come from the heterogeneity within the class and the existence of urban green spaces. Overall, the pronounced improvements on these indiscernible pairs (e.g., grapes-almonds-pistachios, developed-grass/pasture) provide evidence that the discriminative ability of features learned through pre-training is greatly improved.

5.4. Evaluation of pre-trained features of SITS-Former

In this set of experiments, we evaluate the quality of features learned by the pre-trained SITS-Former without fine-tuning. We directly fed the features extracted by the pre-trained SITS-Former to a RF classifier and then optimized the classifier on task data (the pre-trained parameters of SITS-Former were frozen). In this case, features were only learned from unlabeled data without introducing task-specific priors. We call this approach 'RF(SITS-Former$_{pretrained}$)'. In the controlled experiments, we used a randomly initialized SITS-Former as the feature extractor and also trained a RF classifier on task data. We call this approach 'RF(SITS-Former$_{random}$)'. In addition, we directly applied a RF classifier to the concatenation of flattened image patches over the whole time series as the baseline (RF-baseline). Tables 11 and 12 report the accuracy evaluation results on each dataset and Tables 13 and 14 present the per-class F1-scores on test sets. The corresponding confusion matrices are depicted in Figs. 9–10.

For both datasets, RF(SITS-Former$_{pretrained}$) performs significantly better than the other two competing methods. Specifically, RF(SITS-Former$_{pretrained}$) outperforms the RF baseline by 3.53% OA / 2.79% AA on the California dataset, and 0.54% OA / 1.33% AA on the Missouri dataset. The performance gain on the Missouri dataset is less than that achieved on the California dataset, owning to the difference in data distribution between the pre-training data and the task data. The F1-scores of almost all classes have improved using pre-trained features, including classes that rarely appear in the pre-training set, such as soybeans and woody wetlands. These results indicate that the proposed self-supervised pre-training method can extract high-quality features with good transfer performance. However, the untuned model (RF(SITS-Former$_{pretrained}$)) lags behind the fully fine-tuned model (SITS-Former$_{pretrained}$) shown in Tables 6 and 7 with a gap of 2%~4% OA, implying that fine-tuning is critical to improving the performance of pre-trained models on downstream tasks.

5.5. Effect of varying the number of labels

We also investigate the effect of pre-training as the number of labeled samples increases. We trained or fine-tuned each model using varied number of labels from 100 to 1000 per class and a fixed validation set of 1000 samples per class, and then tested all remaining labeled samples. The results are averaged over five random seeds (Fig. 11).

We observe that the pre-trained SITS-Former obviously outperforms all the non-pre-trained models in all cases. On the Missouri dataset, the OA achieved by the pre-trained SITS-Former with 100 labels is comparable to that of its non-pre-trained counterpart achieved with 500 labels. Moreover, the AA achieved by the pre-trained SITS-Former with 100 labels even surpasses those of all the non-pre-trained models using 1000 labels. Among all the pre-trained models, the performance of SITS-Former is consistently better than SITS-BERT and slightly better than CNN-Transformer in most cases, showing that SITS-Former can better exploit the rich information of Sentinel-2 time series.

It is also notable that the accuracy gains brought by pre-training decrease gradually as labeled data grow. Training with 100 labels, the pre-trained SITS-Former yields a performance gain of 2.20%~2.91% OA and 2.66%~3.53% AA over its non-pre-trained counterpart. In contrast, the performance gain using 1000 labels decreases to 0.11%~0.60% OA and 1.01%~1.22% AA, which is still considerable but lower than the difference observed with fewer labels. These results indicate that

Table 10
Per-class F1-score (%) for the Missouri dataset considering different competing methods.

Method	Corn	Cotton	Rice	Soybeans	Fallow	OpenWater	Developed	Forest	Grass/Pasture	WoodyWetlands
RF	93.00	93.70	94.31	89.83	75.05	89.81	41.52	89.68	68.60	84.50
SITS-BERT	92.15	89.97	92.51	88.22	78.02	91.86	44.55	90.70	64.86	81.41
ConvRNN	90.62	87.93	94.83	86.67	76.78	84.48	41.33	90.25	66.70	79.94
DuPLO	93.11	94.87	94.72	91.59	78.09	89.02	44.84	90.50	70.50	82.72
CNN-Transformer	95.72	94.63	96.57	92.79	80.96	94.82	44.48	89.85	70.89	81.94
SITS-Former	95.93	94.67	96.63	93.40	84.13	95.68	46.28	90.50	71.25	81.82
SITS-BERT$_{pretrained}$	96.39	96.26	97.00	94.54	84.33	95.20	54.08	92.09	75.80	85.22
CNN-Transformer$_{pretrained}$	97.22	96.99	**97.58**	95.16	86.16	95.96	54.20	**93.06**	76.01	**88.11**
SITS-Former$_{pretrained}$	**97.33**	**97.34**	97.55	**95.45**	**86.39**	**96.31**	**55.68**	92.46	**77.62**	87.11

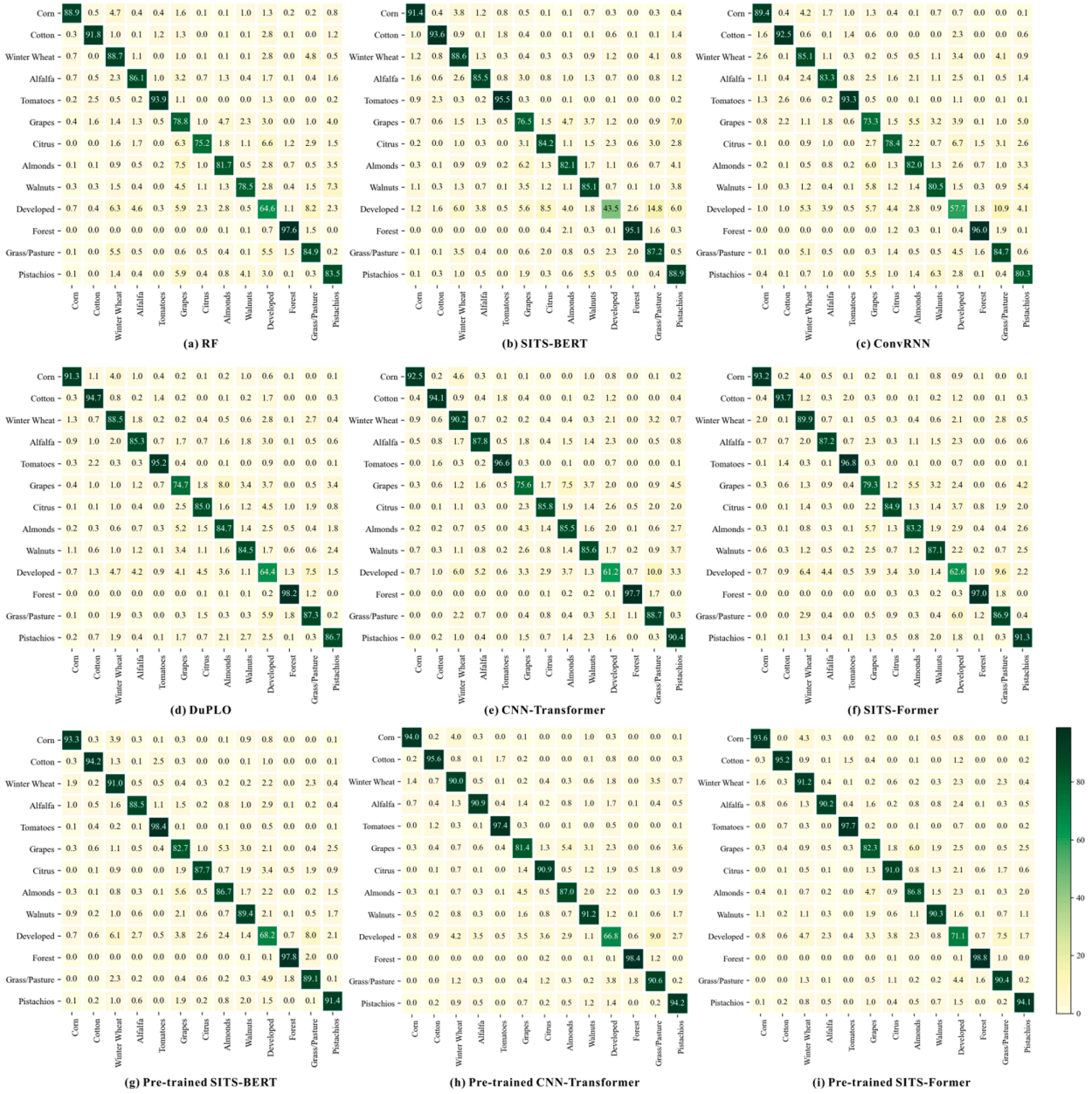


Fig. 7. Confusion matrices obtained on the California dataset by different comparison methods: (a) RF, (b) SITS-BERT, (c) ConvRNN, (d) DuPLO, (e) CNN-Transformer, (f) SITS-Former, (g) pre-trained SITS-BERT, (h) pre-trained CNN-Transformer, (i) pre-trained SITS-Former.

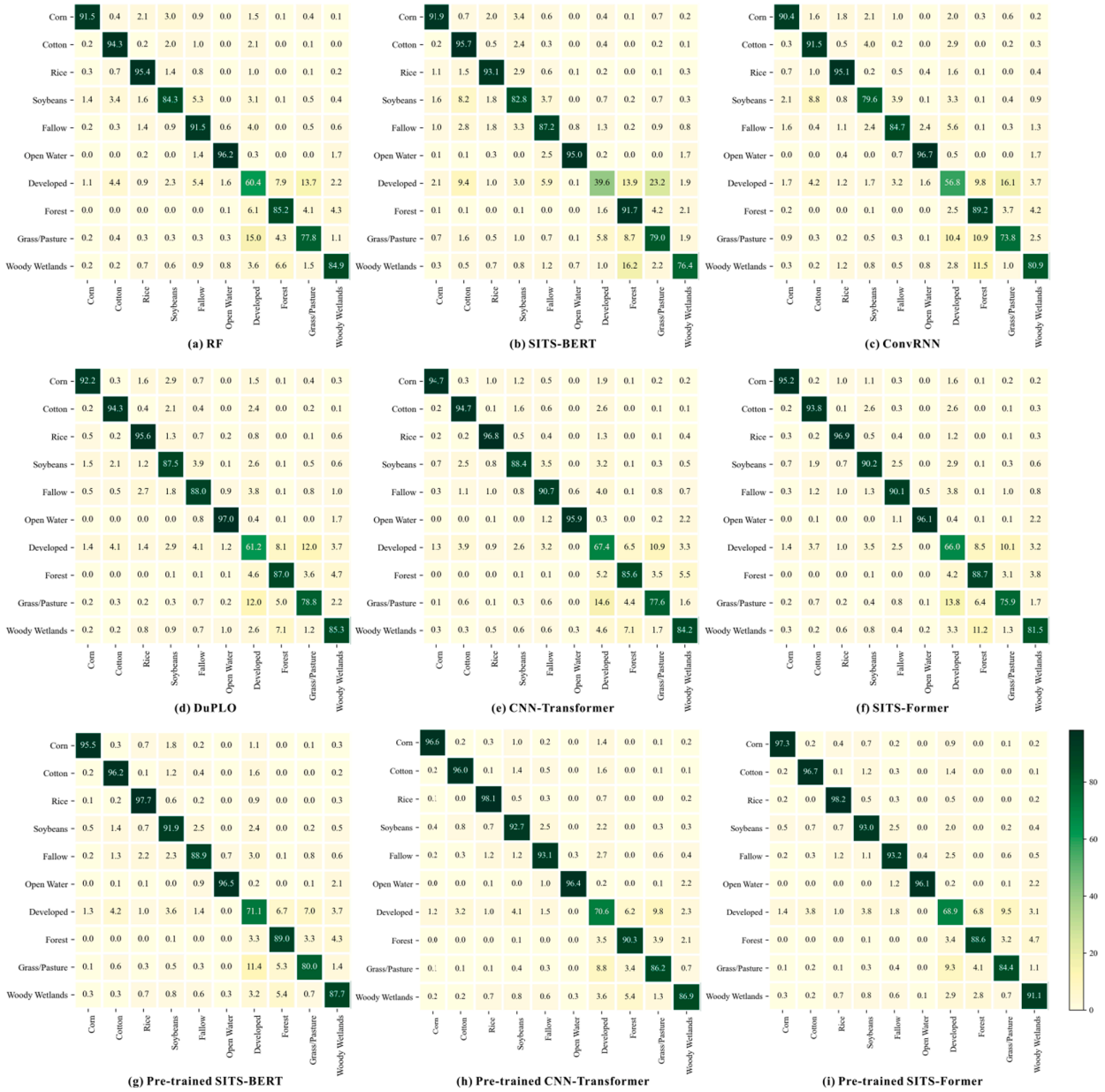


Fig. 8. Confusion matrices obtained on the Missouri dataset by different comparison methods: (a) RF, (b) SITS-BERT, (c) ConvRNN, (d) DuPLO, (e) CNN-Transformer, (f) SITS-Former, (g) pre-trained SITS-BERT, (h) pre-trained CNN-Transformer, (i) pre-trained SITS-Former.

Table 11
Accuracy evaluation of pre-trained features of SITS-Former on the California dataset.

Method	Validation set			Test set		
	OA (%)	AA (%)	Kappa	OA (%)	AA (%)	Kappa
RF-baseline	84.90 ± 0.24	84.90 ± 0.24	0.8364 ± 0.0026	83.26 ± 0.55	84.77 ± 0.36	0.8075 ± 0.0061
RF($\text{SITS-Former}_{\text{random}}$)	66.54 ± 0.83	66.54 ± 0.83	0.6375 ± 0.0090	65.57 ± 1.76	66.90 ± 1.44	0.6092 ± 0.0189
RF($\text{SITS-Former}_{\text{pretrained}}$)	**87.95** ± 0.19	**87.95** ± 0.19	**0.8695** ± 0.0021	**86.79** ± 0.23	**87.56** ± 0.11	**0.8476** ± 0.0027

classification tasks can benefit from pre-training even with a plenty of labeled data, but the performance gains decrease as labeled data increase.

5.6. Processing speed

We compare the processing speed of SITS-Former with other competing methods under the same experimental configuration (Table 15). To be fair, only patch-based methods were compared.

Table 12
Accuracy evaluation of pre-trained features of SITS-Former on the Missouri dataset.

Method	Validation set			Test set		
	OA (%)	AA (%)	Kappa	OA (%)	AA (%)	Kappa
RF-baseline	86.93 ± 0.63	86.93 ± 0.63	0.8548 ± 0.0070	88.62 ± 0.34	86.69 ± 0.22	0.8644 ± 0.0039
RF(SITS-Former$_{random}$)	64.23 ± 1.65	64.23 ± 1.65	0.6026 ± 0.0184	60.46 ± 2.00	64.36 ± 1.20	0.5427 ± 0.0212
RF(SITS-Former$_{pretrained}$)	**88.17** ± 0.33	**88.17** ± 0.33	**0.8685** ± 0.0037	**89.16** ± 0.30	**88.02** ± 0.16	**0.8710** ± 0.0035

Table 13
Per-class F1-score (%) evaluation of pre-trained features of SITS-Former on the California dataset.

Method	Corn	Cotton	WinterWheat	Alfalfa	Tomatoes	Grapes	Citrus	Almonds	Walnuts	Developed	Forest	Grass/ Pasture	Pistachios
RF-baseline	**84.58**	92.46	38.72	85.20	92.15	81.07	80.12	85.79	80.07	51.51	88.68	90.49	81.28
RF(SITS-Former$_{random}$)	37.56	78.20	27.47	71.98	64.65	65.55	48.98	66.74	48.31	43.08	77.91	79.77	61.35
RF(SITS-Former$_{pretrained}$)	77.74	**93.27**	**58.44**	**88.85**	**94.08**	**83.97**	**86.65**	**87.91**	**80.72**	**56.56**	**90.49**	**93.54**	**87.63**

Table 14
Per-class F1-score (%) evaluation of pre-trained features of SITS-Former on the Missouri dataset.

Method	Corn	Cotton	Rice	Soybeans	Fallow	OpenWater	Developed	Forest	Grass/Pasture	Woody Wetlands
RF-baseline	92.93	**94.66**	93.64	90.47	**76.10**	86.23	42.82	90.60	73.44	84.99
RF(SITS-Former$_{random}$)	76.28	75.61	56.28	49.10	35.09	65.75	18.25	76.68	44.27	65.62
RF(SITS-Former$_{pretrained}$)	**96.05**	94.42	**95.85**	**91.07**	69.79	**96.88**	**44.64**	**91.68**	**74.26**	**86.07**

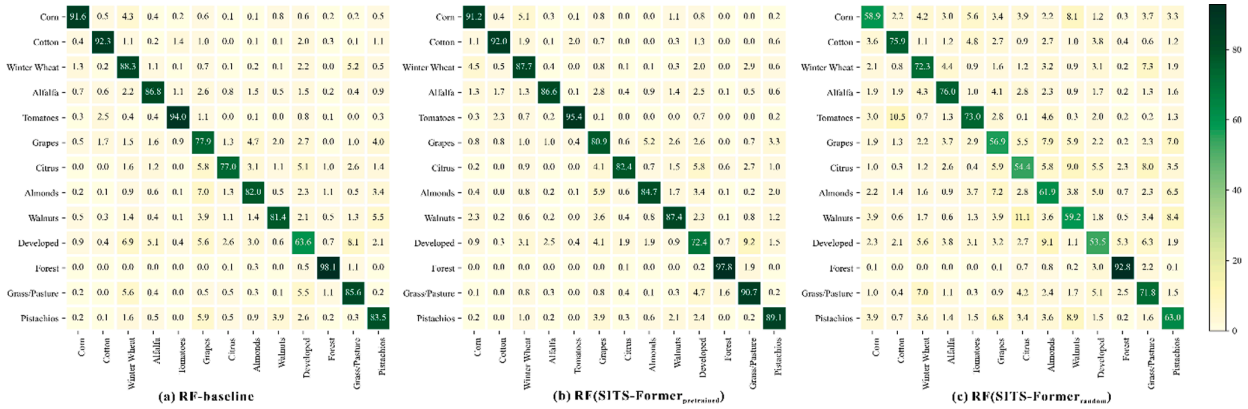


Fig. 9. Confusion matrices obtained on the California dataset by (a) RF-baseline, (b) RF(SITS-Former$_{pretrained}$), (c) RF(SITS-Former$_{random}$).

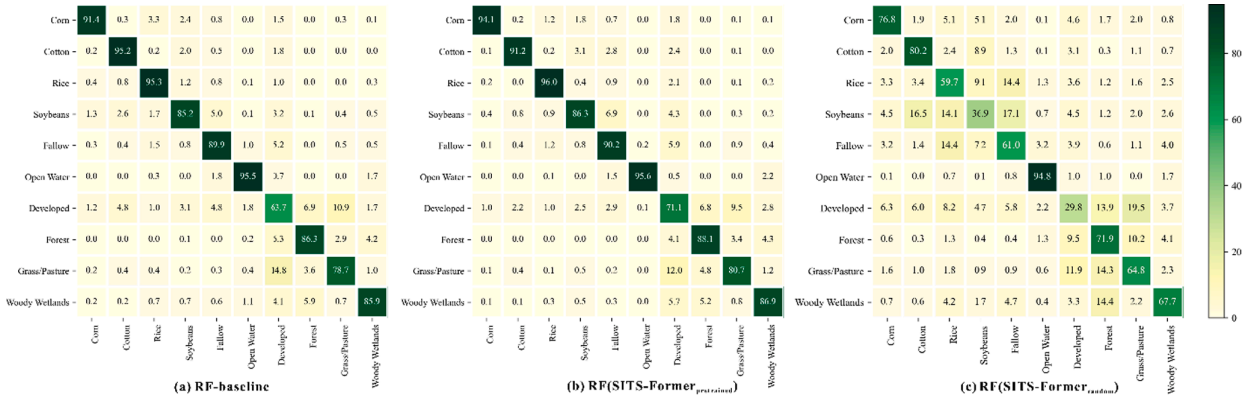


Fig. 10. Confusion matrices obtained on the Missouri dataset by (a) RF-baseline, (b) RF(SITS-Former$_{pretrained}$), (c) RF(SITS-Former$_{random}$).

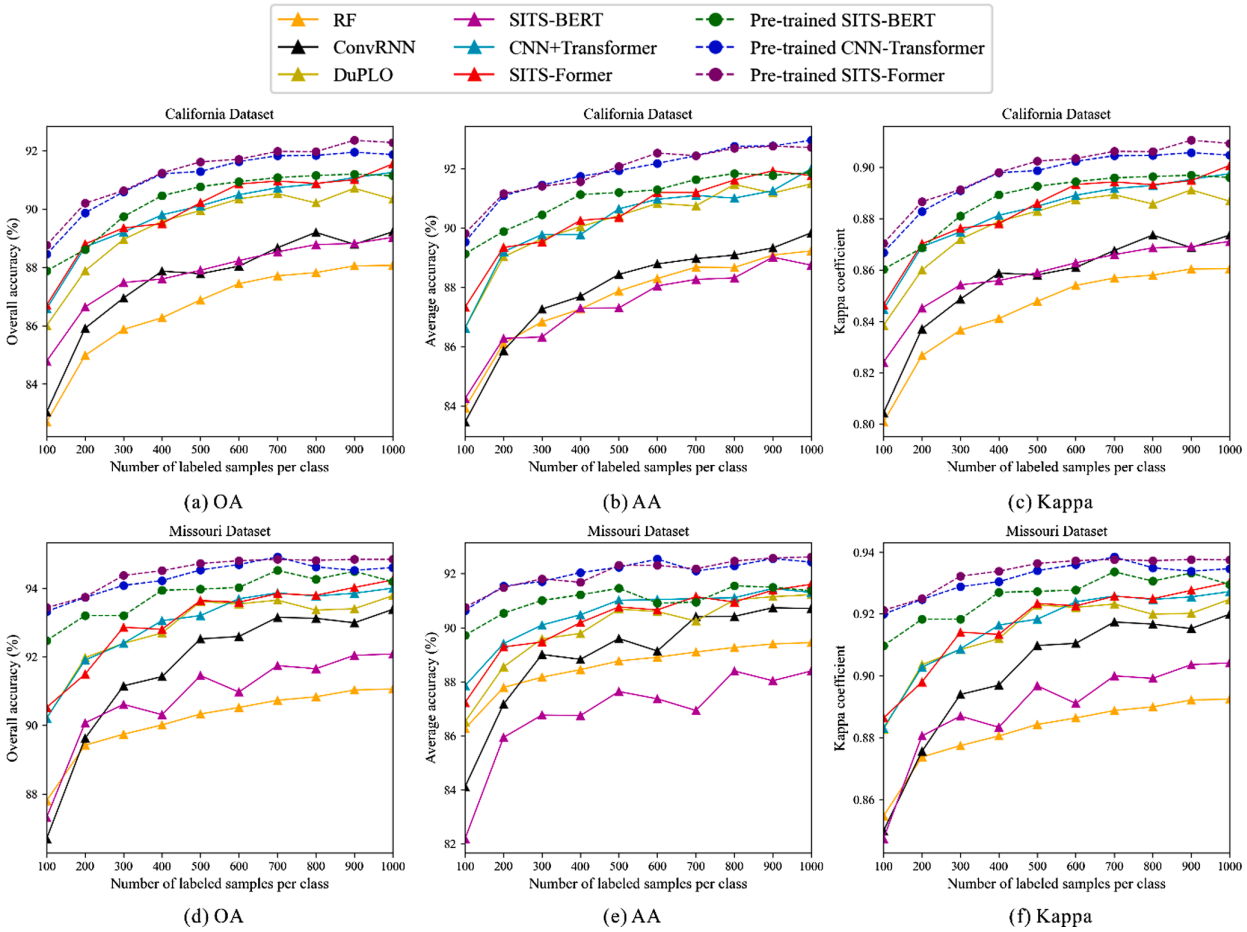


Fig. 11. Accuracy evaluations of different comparison methods using varied number of labels. The first line shows the results on the California dataset, and the second line shows the results on the Missouri dataset: (a)(d) OA, (b)(e) AA, (c)(f) Kappa coefficient.

Table 15
The number of samples processed per second.

Model	Samples per second
ConvRNN	190
DuPLO	352
CNN-Transformer	385
SITS-Former	384

Compared to RNN-based models (ConvRNN and DuPLO), Transformer-based models (SITS-Former and CNN-Transformer) are computationally more efficient. The processing speed of SITS-Former is almost equivalent to that of CNN-Transformer, indicating that the use of 3D convolutions will not cause a significant drop in efficiency. Another interesting finding is that the pre-trained model converges to an optimal solution much faster than a non-pre-trained one (Fig. 12). The optimal number of epochs for fine-tuning SITS-Former is 17 on average, while training the same model from scratch needs at least twice as many epochs. These facts underline that the pre-trained SITS-Former is more suitable for handling large-scale datasets.

6. Discussion

In this paper, we proposed a pre-trained deep learning model called SITS-Former to classify Sentinel-2 time series, which has shown promising results for crop classification in two experimental areas.

The advantages of SITS-Former are as follows. First, SITS-Former allows for the integration of spatial, spectral, and temporal information of Sentinel-2 time series, and is more computationally efficient compared to RNN-based models. In addition, benefitting from pre-training, SITS-Former effectively alleviates the demand for enough labeled data and significantly improves the performance on label-scarce SITS classification tasks. There are also other types of approaches to address the issue of insufficient labels, including weakly supervised learning (Ienco et al., 2020; Wang et al., 2020), semi-supervised learning (Solano-Correa et al., 2019), knowledge distillation (Bazzi et al., 2020), etc. The proposed self-supervised pre-training method in this study can be used as a complementary tool to the existing approaches to ease the burden of data annotation.

However, as a patch-based model, SITS-Former needs to process every pixel individually at inference time, which is very time-consuming when mapping large areas. It is necessary to develop end-to-end models for semantic segmentation of SITS, such as in Garnot and Landrieu (2021), in which all pixels are labeled at once. However, semantic segmentation models also rely on sufficient labeled data to learn pixel classification and spatial arrangement between classes (Interdonato et al., 2019). Designing effective self-supervised proxy tasks for training such models is challenging and we leave this to future work.

7. Conclusion

In this study, we introduced SITS-Former, which is the first pre-

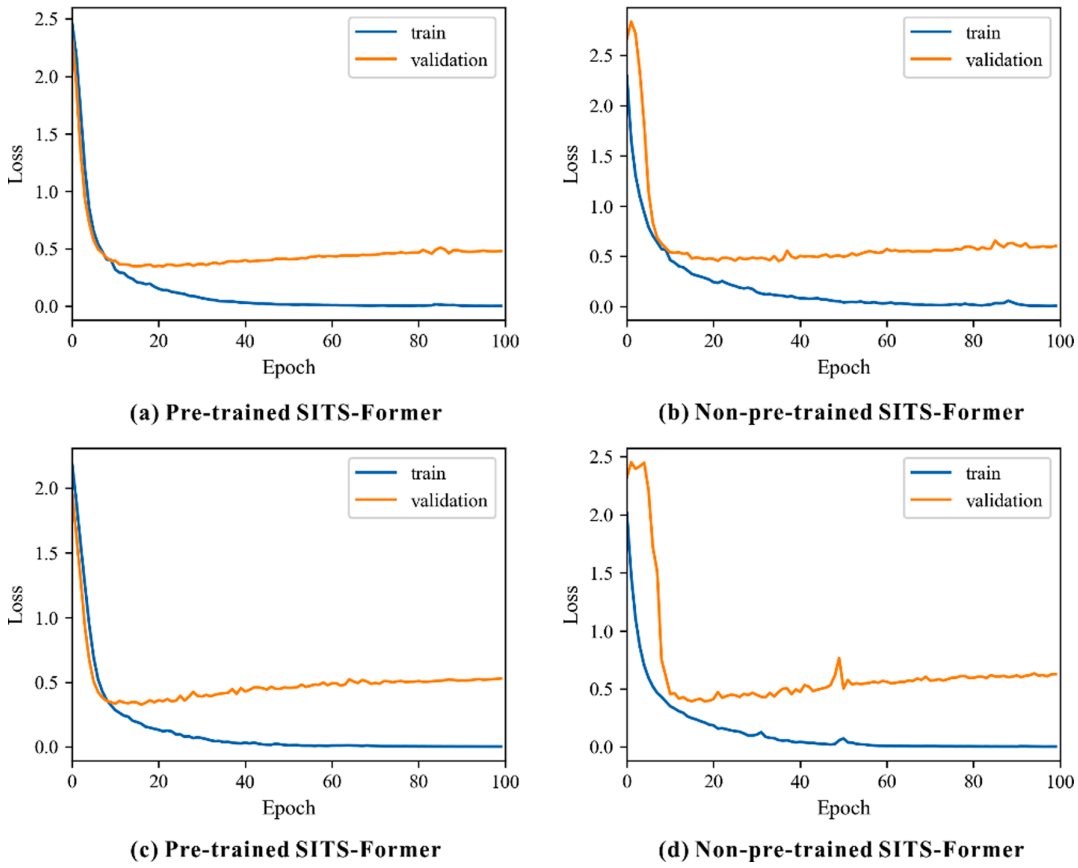


Fig. 12. The first line shows the learning curves of (a) pre-trained SITS-Former and (b) non-pre-trained SITS-Former on the California dataset. The second line shows the learning curves of (c) pre-trained SITS-Former and (d) non-pre-trained SITS-Former on the Missouri dataset.

trained representation model for patch-based SITS classification. SITS-Former is pre-trained on large-scale unlabeled Sentinel-2 time series in a self-supervised manner. It can be easily adapted to a downstream classification task through fine-tuning. Extensive experiments show that SITS-Former outperforms state-of-the-art approaches and offers significant improvements over traditional supervised models, especially when labeled data is scarce. We hope that SITS-Former will make deep learning more accessible to SITS-related applications as it greatly reduces the burden of manual labeling.

Several aspects still need to be explored. For instance, we consider extensions of SITS-Former to fuse Sentinel-1 and Sentinel-2 data as exciting future work. It is also important to explore better proxy tasks to learn more effective representations.

8. Author statement

We declare that this manuscript is original, has not been published before and is not currently being considered for publication elsewhere. We confirm that the manuscript has been read and approved by all the authors and that there are no other persons who satisfied the criteria for authorship but are not listed. We further confirm that the order of authors listed in the manuscript has been approved by all of us.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This research was supported by the National Natural Science Foundation of China (Grant No. 41901356, 61906096, 41701512) and by the Natural Science Foundation of Jiangsu Province under Grant BK20180786.

References

Anwer, R.M., Khan, F.S., van de Weijer, J., Molinier, M., Laaksonen, J., 2018. Binary patterns encoded convolutional neural networks for texture recognition and remote sensing scene classification. ISPRS J. Photogram. Remote Sens. 138, 74–85. https://doi.org/10.1016/j.isprsjprs.2018.01.023.

Bazzi, H., Ienco, D., Baghdadi, N., Zribi, M., Demarez, V., 2020. Distilling Before Refine: Spatio-Temporal Transfer Learning for Mapping Irrigated Areas Using Sentinel-1 Time Series. IEEE Geosci. Remote Sens. Lett. 17 (11), 1909–1913.

Benedetti, P., Ienco, D., Gaetano, R., Ose, K., Pensa, R.G., Dupuy, S., 2018. M3Fusion: A Deep Learning Architecture for Multiscale Multimodal Multitemporal Satellite Data Fusion. IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens. 11, 4939–4949. https://doi.org/10.1109/Jstars.2018.2876357.

Boryan, C., Yang, Z., Mueller, R., Craig, M., 2011. Monitoring US agriculture: the US Department of Agriculture, National Agricultural Statistics Service, Cropland Data Layer Program. Geocarto Int 26 (5), 341–358. https://doi.org/10.1080/10106049.2011.562309.

Chen, T., Kornblith, S., Norouzi, M., Hinton, G., 2020. A Simple Framework for Contrastive Learning of Visual Representations, 37th International Conference on Machine Learning. International Machine Learning Society (IMLS), Virtual, Online, pp. 1575-1585.

Cheng, G., Yang, C., Yao, X., Guo, L., Han, J., 2018. When Deep Learning Meets Metric Learning: Remote Sensing Image Scene Classification via Learning Discriminative CNNs. IEEE Trans. Geosci. Remote Sens. 56 (5), 2811–2821.

Cheng, G., Zhou, P., Han, J., 2016. Learning Rotation-Invariant Convolutional Neural Networks for Object Detection in VHR Optical Remote Sensing Images. IEEE Trans. Geosci. Remote Sens. 54 (12), 7405–7415.

Devlin, J., Chang, M.-W., Lee, K., Toutanova, K., 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding, 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. Association for Computational Linguistics (ACL), Minneapolis, MN, United states, pp. 4171-4186.

do Nascimento Bendini, H., Garcia Fonseca, L.M., Schwieder, M., Sehn Körting, T., Rufin, P., Del Arco Sanches, I., Leitão, P.J., Hostert, P., 2019. Detailed agricultural land classification in the Brazilian cerrado based on phenological information from dense satellite image time series. Int. J. Appl. Earth Obs. Geoinf. 82, 101872. https://doi.org/10.1016/j.jag.2019.05.005.

Dong, H., Ma, W., Wu, Y., Zhang, J., Jiao, L., 2020. Self-Supervised Representation Learning for Remote Sensing Image Change Detection Based on Temporal Prediction. Remote Sens. 12 (11), 1868. https://doi.org/10.3390/rs12111868.

Eudes Gbodjo, Y.J., Ienco, D., Leroux, L., 2020. Toward Spatio-Spectral Analysis of Sentinel-2 Time Series Data for Land Cover Mapping. IEEE Geosci. Remote Sens. Lett. 17 (2), 307–311.

Feng, S., Zhao, J., Liu, T., Zhang, H., Zhang, Z., Guo, X., 2019. Crop Type Identification and Mapping Using Machine Learning Algorithms and Sentinel-2 Time Series Data. IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens. 12 (9), 3295–3306.

Garnot, V.S., Landrieu, L., 2021. Panoptic Segmentation of Satellite Image Time Series with Convolutional Temporal Attention Networks. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), pp. 4872–4881.

Garnot, V.S., Landrieu, L., Giordano, S., Chehata, N., 2019. In: Time-Space Tradeoff in Deep Learning Models for Crop Classification on Satellite Multi-Spectral Image Time Series. IEEE, Yokohama, Japan, pp. 6247–6250. https://doi.org/10.1109/IGARSS.2019.8900517.

Hang, R., Li, Z., Ghamisi, P., Hong, D., Xia, G., Liu, Q., 2020. Classification of Hyperspectral and LiDAR Data Using Coupled CNNs. IEEE Trans. Geosci. Remote Sens. 58 (7), 4939–4950.

Hang, R., Li, Z., Liu, Q., Ghamisi, P., Bhattacharyya, S.S., 2021. Hyperspectral Image Classification With Attention-Aided CNNs. IEEE Trans. Geosci. Remote Sens. 59 (3), 2281–2293.

Ienco, D., Eudes Gbodjo, Y.J., Interdonato, R., Gaetano, R., 2020. Attentive Weakly Supervised land cover mapping for object-based satellite image time series data with spatial interpretation. arXiv e-prints, arXiv:2004.14672.

Ienco, D., Gaetano, R., Dupaquier, C., Maurel, P., 2017. Land Cover Classification via Multitemporal Spatial Data by Deep Recurrent Neural Networks. IEEE Geosci. Remote Sens. Lett. 14 (10), 1685–1689.

Interdonato, R., Ienco, D., Gaetano, R., Ose, K., 2019. DuPLO: A DUal view Point deep Learning architecture for time series classificatiOn. ISPRS J. Photogram. Remote Sens. 149, 91–104. https://doi.org/10.1016/j.isprsjprs.2019.01.011.

Ioffe, S., Szegedy, C., 2015. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift, p. arXiv Preprint arXiv:1502.03167.

Ji, S., Zhang, C., Xu, A., Shi, Y., Duan, Y., 2018. 3D Convolutional Neural Networks for Crop Classification with Multi-Temporal Remote Sensing Images. Remote Sens. 10 (2), 75. https://doi.org/10.3390/rs10010075.

Jing, L., Tian, Y., 2020. Self-supervised Visual Feature Learning with Deep Neural Networks: A Survey. IEEE Trans Pattern Anal Mach Intell 43 (11), 4037–4058. https://doi.org/10.1109/TPAMI.2020.2992393.

Kingma, D.P., Ba, J., 2014. Adam: A method for stochastic optimization, arXiv preprint arXiv:1412.6980.

Kolesnikov, A., Zhai, X.H., Beyer, L., 2019. Revisiting Self-Supervised Visual Representation Learning, Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1920-1929 DOI: 10.1109/Cvpr.2019.00202.

Lebourgeois, V., Dupuy, S., Vintrou, É., Ameline, M., Butler, S., Bégué, A., 2017. A Combined Random Forest and OBIA Classification Scheme for Mapping Smallholder Agriculture at Different Nomenclature Levels Using Multisource Data (Simulated Sentinel-2 Time Series, VHRS and DEM). Remote Sens. 9 (3), 259. https://doi.org/10.3390/rs9030259.

Lei Ba, J., Kiros, J.R., Hinton, G.E., 2016. Layer Normalization. arXiv e-prints, arXiv: 1607.06450.

Li, H., Zhang, C.e., Zhang, S., Atkinson, P.M., 2020a. Crop classification from full-year fully-polarimetric L-band UAVSAR time-series using the Random Forest algorithm. Int. J. Appl. Earth Obs. Geoinf. 87, 102032. https://doi.org/10.1016/j.jag.2019.102032.

Li, W.Y., Chen, H., Shi, Z.W., 2021. Semantic Segmentation of Remote Sensing Images With Self-Supervised Multitask Representation Learning. IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens. 14, 6438–6450. https://doi.org/10.1109/Jstars.2021.3090418.

Li, Z., Chen, G., Zhang, T., 2020b. A CNN-Transformer Hybrid Approach for Crop Classification Using Multitemporal Multisensor Images. IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens. 13, 847–858.

Liu, W., Gopal, S., Woodcock, C.E., 2004. Uncertainty and Confidence in Land Cover Classification Using a Hybrid Classifier Approach. Photogramm Eng Remote Sensing 70 (8), 963–971.

Liu, X., Zhang, F., Hou, Z., Mian, L., Wang, Z., Zhang, J., Tang, J., 2021. Self-supervised Learning: Generative or Contrastive. IEEE Transactions on Knowledge and Data Engineering 1–1. https://doi.org/10.1109/tkde.2021.3090866.

Minaee, S., Kalchbrenner, N., Cambria, E., Nikzad, N., Chenaghlu, M., Gao, J., 2021. Deep Learning–based Text Classification. ACM Comput Surv 54 (3), 1–40. https://doi.org/10.1145/3439726.

Misra, G., Cawkwell, F., Wingler, A., 2020. Status of Phenological Research Using Sentinel-2 Data: A Review. Remote Sens. 12 (17), 2760. https://doi.org/10.3390/rs12172760.

Pelletier, C., Valero, S., Inglada, J., Champion, N., Dedieu, G., 2016. Assessing the robustness of Random Forests to map land cover with high resolution satellite image time series over large areas. Remote Sens. Environ. 187, 156–168. https://doi.org/10.1016/j.rse.2016.10.010.

Pelletier, C., Valero, S., Inglada, J., Champion, N., Marais Sicre, C., Dedieu, G., 2017. Effect of Training Class Label Noise on Classification Performances for Land Cover Mapping with Satellite Image Time Series. Remote Sens. 9 (2), 173. https://doi.org/10.3390/rs9020173.

Pelletier, C., Webb, G., Petitjean, F., 2019. Temporal Convolutional Neural Network for the Classification of Satellite Image Time Series. Remote Sens. 11 (5), 523. https://doi.org/10.3390/rs11050523.

Perez, L., Wang, J., 2017. The Effectiveness of Data Augmentation in Image Classification using Deep Learning. arXiv e-prints, arXiv:1712.04621.

Phiri, D., Simwanda, M., Salekin, S., Nyirenda, V., Murayama, Y., Ranagalage, M., 2020. Sentinel-2 Data for Land Cover/Use Mapping: A Review. Remote Sens. 12 (14), 2291. https://doi.org/10.3390/rs12142291.

Qiu, XiPeng, Sun, TianXiang, Xu, YiGe, Shao, YunFan, Dai, N., Huang, XuanJing, 2020. Pre-trained models for natural language processing: A survey. Sci. China Technol. Sci. 63 (10), 1872–1897. https://doi.org/10.1007/s11431-020-1647-3.

Rußwurm, M., Körner, M., 2017. Temporal Vegetation Modelling Using Long Short-Term Memory Networks for Crop Identification from Medium-Resolution Multi-spectral Satellite Images. In: Proceedings of the 30th IEEE Conference on Computer Vision and Pattern Recognition Workshops, pp. 1496–1504. https://doi.org/10.1109/CVPRW.2017.193.

Rußwurm, M., Körner, M., 2018. Multi-Temporal Land Cover Classification with Sequential Recurrent Encoders. ISPRS Int. J. Geo-Inform. 7 (4), 129. https://doi.org/10.3390/ijgi7040129.

Rußwurm, M., Körner, M., 2020. Self-attention for raw optical Satellite Time Series Classification. ISPRS J. Photogram. Remote Sens. 169, 421–435. https://doi.org/10.1016/j.isprsjprs.2020.06.006.

Saha, S., Bovolo, F., Bruzzone, L., 2020. Change Detection in Image Time-Series Using Unsupervised LSTM. IEEE Geosci. Remote Sens. Lett. 1–5 https://doi.org/10.1109/lgrs.2020.3043822.

Salehi Shahrabi, H., Ashourloo, D., Moeini Rad, A., Aghighi, H., Azadbakht, M., Nematollahi, H., 2020. Automatic silage maize detection based on phenological rules using Sentinel-2 time-series dataset. Int. J. Remote Sens. 41 (21), 8406–8427. https://doi.org/10.1080/01431161.2020.1779377.

Sheeren, D., Fauvel, M., Josipović, V., Lopes, M., Planque, C., Willm, J., Dejoux, J.-F., 2016. Tree Species Classification in Temperate Forests Using Formosat-2 Satellite Image Time Series. Remote Sens. 8 (9), 734. https://doi.org/10.3390/rs8090734.

Solano-Correa, Y.T., Bovolo, F., Bruzzone, L., 2019. In: A Semi-Supervised Crop-Type Classification Based on Sentinel-2 NDVI Satellite Image Time Series And Phenological Parameters. IEEE, Yokohama, Japan, pp. 457–460. https://doi.org/10.1109/IGARSS.2019.8897922.

Solano-Correa, Y.T., Bovolo, F., Bruzzone, L., Fernandez-Prieto, D., 2020. A Method for the Analysis of Small Crop Fields in Sentinel-2 Dense Time Series. IEEE Trans. Geosci. Remote Sens. 58 (3), 2150–2164.

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., Salakhutdinov, R., 2014. Dropout: A Simple Way to Prevent Neural Networks from Overfitting. J. Mach. Learn. Res. 15, 1929–1958.

Sun, Z., Di, L., Fang, H., 2019. Using long short-term memory recurrent neural network in land cover classification on Landsat and Cropland data layer time series. Int. J. Remote Sens. 40 (2), 593–614. https://doi.org/10.1080/01431161.2018.1516313.

Tao, C., Qi, J., Lu, W., Wang, H., Li, H., 2020. Remote Sensing Image Scene Classification With Self-Supervised Paradigm Under Limited Labeled Samples. IEEE Geosci. Remote Sens. Lett. 1–5 https://doi.org/10.1109/LGRS.2020.3038420.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L., Polosukhin, I., 2017. Attention Is All You Need. In: 31st Annual Conference on Neural Information Processing Systems, pp. 5999–6009.

Vincenzi, S., Porrello, A., Buzzega, P., Cipriano, M., Fronte, P., Cuccu, R., Ippoliti, C., Conte, A., Calderara, S., 2021. The color out of space: learning self-supervised representations for Earth Observation imagery. In: Proceedings of the 25th International Conference on Pattern Recognition, pp. 3034–3041. https://doi.org/10.1109/ICPR48806.2021.9413112.

Waldner, F., Diakogiannis, F.I., 2020. Deep learning on edge: Extracting field boundaries from satellite images with a convolutional neural network. Remote Sens. Environ. 245, 111741. https://doi.org/10.1016/j.rse.2020.111741.

Wang, H., Zhao, X., Zhang, X., Wu, D., Du, X., 2019. Long Time Series Land Cover Classification in China from 1982 to 2015 Based on Bi-LSTM Deep Learning. Remote Sens. 11 (14), 1639. https://doi.org/10.3390/rs11141639.

Wang, M.o., Wang, J., Chen, L.i., 2020. Mapping Paddy Rice Using Weakly Supervised Long Short-Term Memory Network with Time Series Sentinel Optical and SAR Images. Agriculture 10 (10), 483. https://doi.org/10.3390/agriculture10100483.

Wang, S., Quan, D., Liang, X., Ning, M., Guo, Y., Jiao, L., 2018. A deep learning framework for remote sensing image registration. ISPRS J. Photogram. Remote Sens. 145, 148–164. https://doi.org/10.1016/j.isprsjprs.2017.12.012.

Wang, Y.A., Chen, Y.N., 2020. What Do Position Embeddings Learn? An Empirical Study of Pre-Trained Language Model Positional Encoding. In: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, pp. 6840–6849.

Yuan, Y., Lin, L., 2021. Self-Supervised Pretraining of Transformers for Satellite Image Time Series Classification. IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens. 14, 474–487.

Yuan, Y., Lin, L., Huo, L.-Z., Kong, Y.-L., Zhou, Z.-G., Wu, B., Jia, Y., 2020. Using An Attention-Based LSTM Encoder–Decoder Network for Near Real-Time Disturbance Detection. IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens. 13, 1819–1832.

Yue, J., Fang, L., Rahmani, H., Ghamisi, P., 2021. Self-Supervised Learning With Adaptive Distillation for Hyperspectral Image Classification. IEEE Trans. Geosci. Remote Sens. 60, 1–13. https://doi.org/10.1109/TGRS.2021.3057768.

Zhang, M., Li, W., Du, Q., 2018. Diverse Region-Based CNN for Hyperspectral Image Classification. IEEE Trans. Image Process 27 (6), 2623–2634.
Zhao, W., Qu, Y., Chen, J., Yuan, Z., 2020a. Deeply synergistic optical and SAR time series for crop dynamic monitoring. Remote Sens. Environ. 247, 111952. https://doi.org/10.1016/j.rse.2020.111952.
Zhao, Z., Luo, Z.e., Li, J., Chen, C., Piao, Y., 2020b. When Self-Supervised Learning Meets Scene Classification: Remote Sensing Scene Classification Based on a Multitask Learning Framework. Remote Sens. 12 (20), 3276. https://doi.org/10.3390/rs12203276.
Zhong, L., Hawkins, T., Biging, G., Gong, P., 2011. A phenology-based approach to map crop types in the San Joaquin Valley. California. Int. J. Remote Sens. 32 (22), 7777–7804. https://doi.org/10.1080/01431161.2010.527397.
Zhong, L.H., Hu, L.N., Zhou, H., 2019. Deep learning based multi-temporal crop classification. Remote Sens. Environ. 221, 430–443. https://doi.org/10.1016/j.rse.2018.11.032.
Zhu, Z., Woodcock, C.E., 2014. Continuous change detection and classification of land cover using all available Landsat data. Remote Sens. Environ. 144, 152–171. https://doi.org/10.1016/j.rse.2014.01.011.