IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 1 November 2022, accepted 13 December 2022, date of publication 15 December 2022,
date of current version 23 December 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3229589

RESEARCH ARTICLE

Paddy Rice Mapping Using a Dual-Path Spatio-Temporal Network Based on Annual Time-Series Sentinel-2 Images

HUI WANG[1,2], BO ZHAO[1,2], PANPAN TANG[1,2], YUXIANG WANG[3], HAOMING WAN[1,2], SHI BAI[1,2], AND RONGHAO WEI[4]

[1]Research Center of Big Data Technology, Nanhu Laboratory, Jiaxing 314002, China
[2]Advanced Institute of Big Data, Beijing 100093, China
[3]PIESAT Information Technology Company Ltd., Beijing 100195, China
[4]Zhejiang Institute of Hydraulics & Estuary, Hangzhou 310020, China

Corresponding author: Panpan Tang (tangpp@nanhulab.ac.cn)

This work was supported in part by the Nonprofit Research Project of Jiaxing City under Grant 2022AY30001, and in part by the Inner Research Fund of Nanhu Laboratory.

ABSTRACT Paddy rice is one of the main foods of the global population. To guarantee paddy rice acreage is essential to ensure food security. Currently, techniques for large-area paddy field mapping rely mainly on complex rule-based machine learning algorithms. But it is difficult for them to achieve an optimal balance between discriminability and robustness. In this article, we proposed a novel deep learning-based approach for large-scale paddy rice mapping, termed dual-path interactive network (DPIN). An annual time-series Sentinel-2 remote sensing images are used as data source. Taking several areas of interest over the middle and lower Yangtze River plain of China as experimental fields, our model achieves an F1-score of 91.22% on the test dataset, which is 1.09% higher than the existing state-of-the-art predictive model, and its inference speed is 1.18 times faster than it. DPIN-Lite is a lightweight variant of DPIN, and while keeping a competitive mapping accuracy, its inference speed is 1.91 times faster than the compared method (with the best score except for DPIN and DPIN-Lite).

INDEX TERMS Paddy rice mapping, deep learning spatio-temporal, sentinel-2.

I. INTRODUCTION

Paddy rice is a staple food for more than half of the global population, and ensuring its production is a great guarantee of global food security and environmental sustainability [1], [2], [3]. According to the statistics from the Food and Agriculture Organization of the United States [4], Asia is the main source of paddy rice worldwide, accounting for about 90% of the global production during the years between 2013 to 2020 [5]. Among them, China alone accounts for 28%. In recent decades, the demand for rice has dramatically increased with population growth [6] and accelerated urbanization. Unfortunately, human activities like high emissions [7] have raised the threat to food supply [8], [9], and in recent years global pandemics and regional warfare greatly increase the risk of food crisis. As for China, due to lower economic benefits from growing staple crops like rice, the non-food phenomenon of cultivated lands is increasingly severe. So, it is highly necessary to accurately extract the paddy rice planting area on a greater scale.

Various methods have been proposed for paddy rice mapping, and remote sensing-based technology has proven to be the most effective [10], [11], [12], [13]. Compared to SAR (Synthetic Aperture Radar) images [14], [15], optical remote sensing images are susceptible to tillage because of their relatively high spectral resolution (from visible to near-infrared, and to short wave infrared) and spatial resolution [16]. Therefore, optical images are still the main data source for paddy rice identification and mapping. Common candidates include Moderate Resolution Imaging Spectroradiometer (MODIS), Landsat-7/8 [17] and Sentinel-2 [18], [19], Planet series, GaoFen-2 and WorldView-3, whose spatial resolutions

The associate editor coordinating the review of this manuscript and approving it for publication was Diego Oliva.

ranging from 250 m to about 0.5 m. To facilitate the requirements of large area cover, and low cost while holding accuracy, the Sentinel-2 imagery with a spatial resolution of 10m and a revisit interval of 5 days was used in the present study.

Paddy rice mapping algorithms can be divided into two categories: phenology-based and spectral learning-based methods, and they can be used in combination. Phenology-based methods use the growth condition indexes of crops to extract their phonologic information and then identify them according to experts' rules or empirical thresholds. The vegetation/water indexes, like Normalized Difference Vegetation Index (NDVI), Enhanced Vegetation Index (EVI), Normalized Difference Water Index (NDWI), etc., are often used [20], [21], [22]. Using them can measure the growing period, such as Start of the Season (SOS), End of the Season (EOS) and Length of the Season (LOS) [23], [24], [25], which are useful for distinguishing paddy rice from other crops or ground features. More often, the transplanting period of rice is monitored because the characteristics of plant, soil and water are blended during this time and paddy rice fields show biggest differences from other objects in the indexes [26], [27]. However, prior planting information are needed, and cloud occlusion would greatly affect in the key period. In addition, studies have proved that since the images in a single phenological period cannot fully reflect the characteristics of the whole rice growth cycle, using images of multiple phenological periods can achieve better results [26], [27].

Spectral learning-based methods can be further divided into methods based on spectral matching, methods based on machine learning, and methods based on deep learning techniques. Spectral matching methods identify rice by measuring the similarity with spectral features or second-order features (NDVI, etc.) [28], [29]. This requires the pre-extracted rice features to be accurate enough, which is hard especially when there is more than one rice ripe pattern. The machine learning methods are often used to learn the spectral characteristic, and then make the prediction. Support vector machines (SVM) [16], random forests (RF) [26], [30] and decision trees (DT) [31] have made good contributions in the field of rice mapping. However, due to their limited learning ability, machine learning methods are usually performed on indexes extracted by phenology methods rather than the complete spectrum channels, so the mapping accuracy is restricted.

Deep learning (DL) models have stronger learning capabilities and can learn expressions from the full spectrum channels. In recent years, some studies have applied DL techniques to paddy rice mapping, including classical convolutional networks [23], [32] and time-based models (like LSTM) [33]. However, convolutional networks lack constraints on temporal sequential relationships, and time-based models ignore spatial dependence. For the paddy rice mapping task, both the temporal variation and spatial texture information are critical, thus, spatio-temporal models are preferred.

Spatio-temporal models have been practiced in other fields and demonstrated their powerful capabilities. One classic model is called ConvLSTM [34], [35], which was first used for predicting the weather, and then for traffic, behavior recognition, anomaly monitoring, and a series of spatio-temporal problems. Later, two improved versions called U-ConvLSTM [36] and U-BiConvLSTM [37] were proposed to improve the training and inference speed, at the cost of the ability to capture temporal features. In [38], the spatio-temporal model structure was explored with the time-series Sentinel-2 images for the crop segmentation task, and finally, a model named UTAE with the highest recognition accuracy was proposed. It is one of the currently recognized spatio-temporal models that can learn the features of remotely sensed time-series images. Its structure is very enlightening for our work, but still has some problems: Insufficient fusion of spatial and temporal features, large memory consumption and slow inference speed, etc.

Therefore, in this paper, we aim to mapping paddy rice in high precision using time series Sentinel-2 images, and contributions are: (1) Designing a better spatio-temporal DL model, which would enhance the fusion of spatial and temporal features, reduce memory consumption and improve model inference speed; (2) Proposing a paddy rice mapping scheme including data selection, data pre-processing and model training. The paper is organized as follows: Section 2 introduces the study area and methods; section 3 presents the mapping results and relevant analysis. Section 4 discusses the limitation and some explorations of our work; Section 5 is the concluding chapter.

II. MATERIALS

A. STUDY AREA

The middle and lower Yangtze River plain is one of the most prominent rice-producing areas in China. Its total area is about 200,000 km2, and spans seven provinces, including Hubei, Hunan, Jiangxi, Anhui, Jiangsu, Zhejiang and Shanghai. The plain is somewhat swampy, made up of many lakes and rivers, making it suitable for rice growing and freshwater fish, and it is therefore known as the ''land of fish and rice.'' Controlled by subtropical monsoon climate, the plain is rich in rainfall, with an average annual precipitation of 1000∼1500 mm. According to rough estimates, the rainfall days account for about one-third of a year, and are concentrated between April and June. Sometimes influenced by typhoons, there is also plenty of rainy days from July to September.

Five areas of interest (AOIs), denoted as AH1, AH2, AH3, JX1 and ZJ1 (see Fig.1), respectively, were chosen to act as our experimental fields. However, due to confidentiality agreements, we are not able to add more location details. According to the ''Dataset of Rice Spatial Distribution in Nine Provinces and One City in Southeast China,'' the paddy rice in these AOIs is ripe once a year, and some are in rotation with other dry crops such as wheat, maize, etc. According to

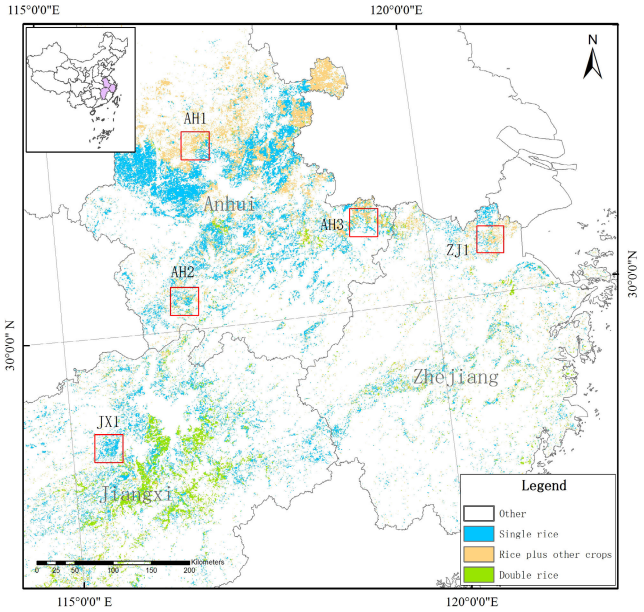

FIGURE 1. Paddy rice distribution of five sample areas (marked with red) in the Middle and Lower Yangtze River. The five sample areas are denoted as AH1, AH2, AH3, ZJ1 and JX1, respectively. The base map is quoted from the paddy rice distribution product in 2013 in the "Dataset of Rice Spatial Distribution in Nine Provinces and One City in Southeast China" [39].

the transplanting time, paddy rice can be divided into three types: early-season rice, mid-season rice and late-season rice. The early-season rice is usually transplanted in late March to early April and harvested in mid-to-late July. The mid-season rice is usually transplanted from early April to late May and harvested in mid-to-late September. The late-season rice is usually transplanted in mid-to-late June and harvested in early-to-mid October. The paddy rice grown in the rotation is mainly mid or late-season rice.

B. DATASET

Multi-temporal Sentinel 2-SR images falling within the AOIs were collected based on Google Earth Engine (GEE). The time span of these images ranges between January and December 2019, and their cloud coverage is kept $\leq 9\%$. Three atmospheric spectral channels (1, 9 and 10) of the Sentinel-2 satellite were routinely removed from the data while retaining the other 10 channels. Finally, these images were resampled at a spatial resolution of 10m per pixel. The ground truth image of paddy rice was manually annotated by agricultural image interpretation experts based on google earth images and field investigations.

Beyond resampling, a series of image pre-processing operations were performed, which are described in detail in Section 2.3. And then, all the collected images were cropped into small patches using a sliding window with a stride of 128 non-overlapping pixels. After that, the patches (or say samples) without paddy fields were removed from the dataset, and the remaining 6280 samples were used for subsequent model training and prediction. Based on the Python platform, each sample in the dataset was stored

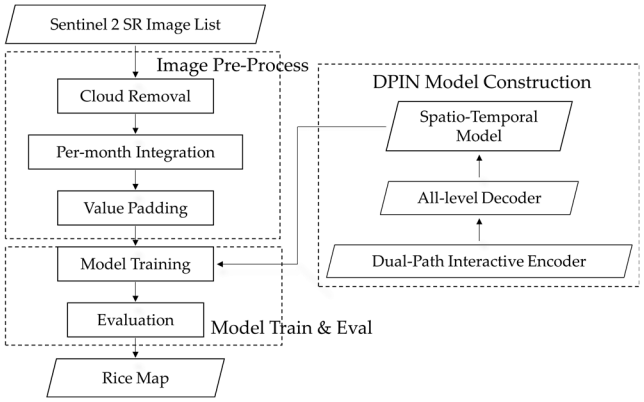

FIGURE 2. Workflow of paddy rice mapping. Including three parts: image pre-process, model design and model operations.

as multidimensional arrays: [T, 10, 128, 128] in ".h5py" format, where T is the length of the time series, 10 is the number of channels, 128 & 128 denote the height and width. In our experiment, 80% of the samples were used for training and the remaining 20% for validation. Finally, a five-fold validation method was used to obtain the paddy-field map over the entire sample area.

III. MATHOLOGY

A. WORKFLOW

Fig. 2 displays the workflow of the proposed method for paddy-field mapping, which consists of three procedures/modules: image pre-processing, construction of the DPIN model, model training and evaluation. First, the pre-processing module consists of cloud removal, per-month image integration, and zero-padding for absent time-series images. Second, the DPIN model is composed of two submodules: a dual-path interactive encoder (DPIE) and an all-level decoder (ALD). Third, for model training: considering lightweight parameters and sufficient training samples, we trained our DPIN model from scratch; for model evaluation: a five-fold validation strategy is applied to evaluate the accuracy and reliability of the proposed method.

B. IMAGE PRE-PROCESSING

The cloud removal operation is a two-step procedure. The first one is the image-level removal. As mentioned, only the images having cloud coverage $\leq 9\%$ were selected to constitute our training datasets. However, a small cloud-coverage threshold may result in absence of some time-series images. In [40] and [41], the adopted threshold is $\leq 70\%$. However, the accuracy loss caused by these cloud-contaminated input images is very serious. To balance the cloud coverage and data deficiency, the empirical value 9% was used. The second one is pixel-level cloud removal. Based on GEE platform, we use the QA60 band, which embedded in Sentinel-2 data flagged the Opaque Clouds pixels (Bit 10) and Cirrus Clouds pixels (Bit 11) [42], to conduct cloud detection and removal. However, the QA60 band is only sensitive to

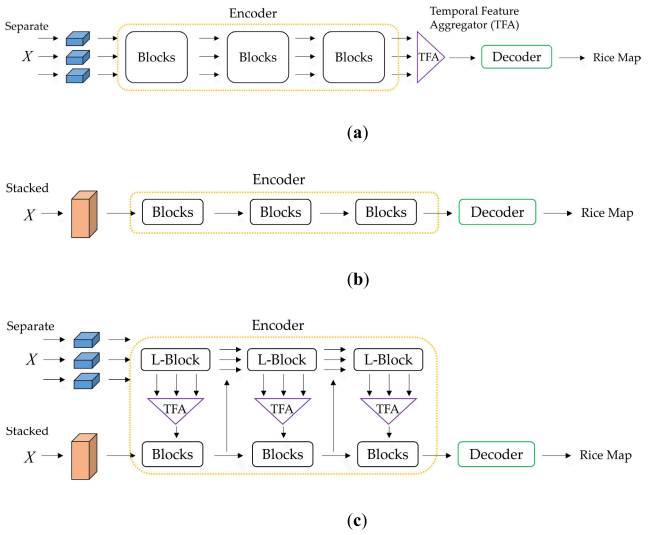

FIGURE 3. Framework of different types of networks: (a) Spatio-temporal network (UTAE format); (b) Spatial network; (c) Dual-path interactive network.

thick clouds, and pixels contaminated by thin clouds still remain.

In theory, the satellites in the Sentinel-2 constellation could provide a revisit time of 5 days in cloud-free conditions. In practice, the data availability in several months (such as June, July and October) can be scarce due to cloud cover. One way to address this issue is to conduct data integration. For example, supposing there are 1 to 6 image(s) available in one month, we take the medians of the pixel values of these images for each pixel to generate an integrated image. By doing this, we minimize the impact of cloud cover – the pixel values contaminated by clouds can be dismissed as outliers during this averaging process. Finally, we can obtain 12 time-series images from January to December by conducting a per-month data-integration operation within the year of 2019.

Even so, there are still some images unavailable, and our solution is twofold as follows: (1) substitute the missing images in 2019 for the counterpart images from the 2018 or 2020 image database; (2) when there are no substitutable images, we conduct zero-padding for the missing data. This operation ensures that the input time series are of the same length. In practice, the operation (1) can deal with most of the described problems, and the operation (2) is not a common case, so it has limited impact on the model performance. Meanwhile, the high-quality rice mapping results we got prove that DPIN has some ability to resist data missing.

C. MODEL CONSTRUCTION

The spatio-temporal models could extract the growth features of paddy rice from the time-series images, and the encoder's consumption of memory is related mostly to the length of the time-series. Fig. 3 (a) is such an example, in which a temporal feature aggregator (TFA) module is constructed to integrate the multi-temporal feature representations derived

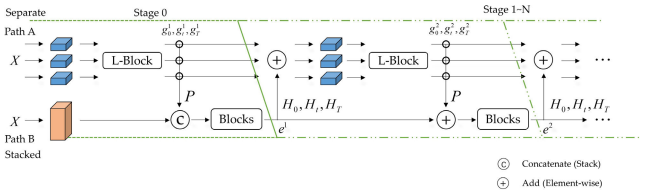

FIGURE 4. Structure of dual-path interactive encoder, consists of separated path, stacked path and interactive branch (symbolled as P and $H_t(t \in [0, T])$) with N stages.

from the encoder, and then, the resulting representations are sent to the decoder to generate the rice distribution map. By contrast, to reduce the memory cost, Fig. 3(b) is designed to use the stacked time-series images as input to the spatial networks, however, it has a difficulty in producing high-quality rice mapping results because the time-sequential context information is not further exploited during the encoding process. To solve these problems, we designed a dual-path spatio-temporal model for paddy rice mapping, termed DPIN, and its core components include a dual-path-interactive encoder (DPIE) and an all-level decoder (ALD), as shown in Fig. 3(c).

DPIE adopts an (N-stage) dual-path structure connected by the interactive modules P and $H_t(t \in [0, T])$. First, the input time-series images are transformed into a four-dimensional tensor X with the shape of [T, C, H, W],where T denotes the length of the time sequence, and in our case T=12, representing the observation data come from 12 months in one year; C denotes the number of channels, and in our case C=10; H and W represent the height and width of each input image. And then, the DPIE module encodes X through two paths: (1) the ''separate'' path: the time-sequential images are encoded by using a spatial convolutional network in a parallel manner, which involves creating a series of L-Blocks; (2) the ''stacked'' path: all the input images are first concatenated and then encoded by a standard convolutional network to integrate the multi-temporal information. The feature representations derived from these two paths will be sent to and integrated by the interactive modules at different stages. The interactive module P plays the role of a bridge between the two paths: at each stage, the feature representation derived from the ''separate'' path is sent to and processed by the interactive module, and then integrated with the corresponding representation derived from the ''stacked'' path. In a similar fashion, $H_t(t \in [0, T])$ facilitates the communication from the ''stacked'' path to the ''separate'' path. Note that: in Fig.4, in order to save memory space, we stipulate that the dimension of the L-Block along the ''separate'' path must be even smaller than that of the Block along the ''stacked'' path.

The architectures of Block and L-Block are shown in Tab. 1 and Tab. 2, respectively. The Block module consists of five convolutional layers from top to bottom, which are: a 7×7 depth-wise convolution layer, a layer normalization layer (see [43]), a GELU activation layer, a 1×1 convolution layer and the other layer normalization layer. Such an architecture

TABLE 1. Structure of the block module.

Module Name	Layer/Operation (Op) Name	Kernel	Pad	Stride	Shortcut
Block	Depth Conv	7×7	3	1	
	Layer Norm	-	-	-	
	GELU	-	-	-	Yes
	Conv	1×1	0	1	
	Layer Norm	-	-	-	

TABLE 2. Structure of the L-block module.

Module Name	Layer/Operation （Op）/Module Name	Kernel	Pad	Stride	Shortcut
L-Block (at stage 0)	Conv	1×1	0	1	No
	Layer Norm	-	-	-	No
	Concat (Op)	-	-	-	No
	Block	-	-	1	Yes
	Reshape (Op)			-	No
L-Block (at stages 1 to N)	Conv	2×2	0	2	No
	Layer Norm	-	-	-	No

The "Concat" operation concatenates the multi-temporal inputs of size [T, C, H, W] in the channel dimension to form a new tensor with size [T×C, H, W] that is readable for Block; The "Reshape" operation is used to reshape the "Block" module's output features from [T×C, H, W] to [T, C, H, W], so that is readable for the next convolutional layer.

is borrowed from the ConvNeXt model [44], and on this basis we developed the L-Block module. To avoid memory overhead, L-Block has two lightweight structures: as shown in Fig. 4, from stage 1 to stage N (N=4), it is designed to have a 2×2 convolution layer and a layer normalization layer; while at stage 0, it consists of a 1×1 convolution layer, a layer normalization layer and a Block module. The input of this Block module is a stacked tensor with size [T×C, H, W], and the output of it is reshaped to fit the style of a normal tensor: [T, C, H, W]. By doing this, we could maintain the inherent time-series context relations and add spatiotemporal information onto the "stacked" path.

The "separate" path works in the following way: For each time point t, the encoder G_l at level l takes as input the feature representations of the previous layer g_t^{l-1}, and outputs a representation map g_t^l of size $c^l \times h^l \times w^l$ with $h^l = h^{l-1}/x$ and $w^l = w^{l-1}/x$, where $x$is 2 when $l \in [2, N]$, x equals 1 when l is 1. The same assignment rule is applicable to the values of and w^l in the decoder and the layers along the "stacked" path. The g_t^l is computed as follows:

$$g_t^l = G_l(g_t^{l-1}) \quad for\ l \in [1, N],\ t \in [0, T] \tag{1}$$

The "stacked" path works in the following way: First, the orange box in Fig. 4 indicates the concatenating operation of X (size: $[T \times c, h, w]$). And then, the encoder E_l at level l takes as input the feature representations of the previous layer e^{l-1}, and outputs a feature representation e^l of size $(T * c^l) \times h_l \times w_l$.Therein, e^l is computed as follows:

$$e^l = E_l(e^{l-1}) \quad for\ l \in [1, N] \tag{2}$$

To facilitate the communication between the temporal and spatial representations, we add several interactive modules in between the "separate" path and the "stacked" path, which

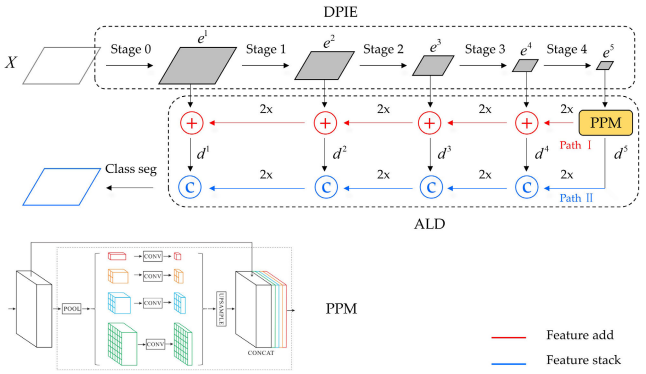

FIGURE 5. Structure of all-level decoder, which contains two paths: path I (marked with red) and path II (marked with blue).The architecture of PPM is after [45].

is noted as P and $H_t(t \in [0, T])$ in Fig. 4, and they are actually all 1×1 convolution layer. As a lightweight TFA, F takes the stacked g_t^l as input ($t \in [0, T]$), and outputs the spatiotemporally fused representations based on the time-series representations. These fused representations are then added or concatenated with e^{l-1} to get e^l. H_t is used to deliver the global spatio-temporal information from the "stacked" path to the "separate" path: it takes e^{l-1} as input, and outputs a feature representation ($H_t(e^{l-1})$) with the same size as g_t^l, and then the multi-temporal results of $g_t^{l-1} + H_t(e^{l-1})$ will be put to the L-block for further processing. Herein, the $H_t(\cdot)$ is used to reduce the feature dimension of e^{l-1}. Given the above, the following formulas describe how the DPEI works:

$$\begin{aligned} g_t^0 &= X_t \\ e^0 &= Conv(X) \\ e^1 &= E_1(e^{l-1} \odot P([g_t^l]_{t=0}^T)) \\ g_t^l &= [G_l(g_t^{l-1} + H_t(e^{l-1}))]_{t=0}^T \quad for\ l \in [1, N] \\ e^l &= E_l(e^{l-1} + P([g_t^l]_{t=0}^T)) \quad for\ l \in [2, N] \end{aligned} \tag{3}$$

where $[\cdot]_{t=0}^T$ denotes the stacked stack time-series features from t_0 to t_T, $\odot$ means the concatenation operation, X denotes the time-sequential images, and $Conv(\cdot)$ is a convolution module including convolution layer, a normalization layer and a GELU layer.

In accordance with the encoder, our decoder also has two paths. As shown in Fig. 5, as low-level features could preserve the edge details, we created a parallel path (path II, the blue line) to concatenate the feature representations derived from DPIE at different scales. However, due to the shallow network projection, low-level features usually lack of global spatial information which is helpful for high-accuracy classification. Inasmuch, we added a PPM (Pyramid Pooling Module, see reference [45] for details) module at the end of the encoder to extract the deepest spatiotemporal representations. After that, we created another path (path I, the red line) to fuse the low-level features and the feature representations e^l derived from the encoder, and the newly generated representations which contain more global and essential spatiotemporal information would be sent to path II for further processing.

This idea comes from [46]. Note that: PPM is incorporated in DPIE because it can adaptively capture the spatiotemporal information at different scales by using four pooling windows of different sizes, thus facilitating the extraction of deeper and meaningful paddy rice semantic features.

Adhere to the path II there are N - 1 stages (i.e., $\mathcal{D}_l$, $l = 1 \ldots N$), and each of them consists of a series of convolutional layers, GELU layers and normalization layers. The output of $\mathcal{D}_l$is a feature representation d^l with size $(T \times c_l) \times h_l \times w_l$, where d^l is computed in the following way:

$$\begin{aligned} d^l &= \mathcal{D}_l(PPM(e^{l+1})) \quad when\ l = N-1 \\ d^l &= \mathcal{D}_l(\mathcal{U}(d^{l+1}) + e^{l+1}) \quad for\ l \in [1, N-2] \end{aligned} \tag{4}$$

where $\mathcal{U}$ denotes the up-sampling operation. And PPM, as shown in Fig. 5b, is used to extract multi-scale spatial features. The pooling scale of PPM is set as (1, 2, 3 & 6), and its output dimension is empirically set as 256. Given the above, the path II can be formulized as:

$$\hat{y} = \mathcal{F}([\mathcal{U}(d^l)]) \quad for\ l \in [1, N] \tag{5}$$

where $\hat{y}$ is the predicted probability, $\mathcal{F}$ represents the output layer composed of a 1 × 1 convolution layer with output channels of 2 (paddy rice or non-paddy rice) and a softmax function.

The loss function used in this paper is cross entropy, formulated as:

$$\mathcal{L} = -(y \log(\hat{y}) + (1-y)\log(1-\hat{y})) \tag{6}$$

where $\mathcal{L}$ symbols the loss, y denotes the ground truth, and $\hat{y}$ denotes the predicted probabilities output from $\mathcal{F}$.

IV. RESULTS

A. IMPLEMENTATION DETAILS

Details of experiments in this paper are presented here, including the working environment, model setting and training tricks. All codes were written in python 3.8 under the Pytorch 1.10.1 framework, and experiments were conducted on a single NVIDIA Tesla V100 with 32 GB of GPU memory. During the training process, the batch size was set to 14, and the initial learning rate was set to 0.007. A start warm strategy was used in the first 4 epochs, with the total training epochs set to 100. Meanwhile, a learning rate decay strategy named cosine decay and two regularization operations - stochastic depth [51] with a drop rate of 0.1 and weight decay with the param set as 1e-4, were adopted. In data augmentation strategy, random vertical and horizontal flips with a probability of 50%, and a random crop with a size of 96, were applied. An AdamW optimizer with default parameters was used for model training and a mix-precision training scheme was adopted.

Unlike the general neural networks with 4 stages, the stage number N of DPIE was set to 5 in the experiment, and the first stage is dedicated to retaining low-level information. In each stage, there is an L-Block and several Blocks. The Block numbers were set to [1, 3, 3, 27, 3] from stage 0 to stage 4, and their corresponding out-channels were set to [96, 96, 192, 384, 768], respectively. The out-channels of the L-blocks were set to [4, 8, 16, 32, 64] for each time point. The initial weights of Blocks from stage 1 to stage 5 were loaded from a pre-trained ConvNeXt-small model. Because the components in stage 0 and L-Blocks were extra designed, they have no pre-trained parameters. The out-channels of the decoder were set to 256 in all stages. This paper gave two model versions, a standard DPIN and a lighter one named DPIN-Lite. The numbers of Blocks in DPIN-Lite were set to [1, 1, 1, 3, 1] for lower memory cost and higher inference speed.

TABLE 3. Mapping performance of DPIN on five sample areas.

Area Name	Precision (10^{-2})	Recall (10^{-2})	IoU (10^{-2})	F1-score (10^{-2})
AH1	90.97	92.87	85.03	91.91
AH2	91.87	89.70	83.10	90.77
AH3	88.48	92.67	82.69	90.53
ZJ1	89.02	90.94	81.76	89.97
JX1	95.13	96.16	91.65	95.64
Overall	90.41	92.04	83.85	91.22

B. MAPPING RESULTS

In this paper, four metrics were used to evaluate the effectiveness and accuracy of the proposed method, which are Precision, Recall, F1-score and IoU (Intersection over Union). The formulas are given as follows:

$$\begin{aligned} Precision &= \frac{TP}{TP+FP} \\ Recall &= \frac{TP}{TP+FN} \\ F1-score &= 2 \times \frac{Precision \times Recall}{Precision + Recall} \\ IoU &= \frac{TP}{TP+FP+FN} \end{aligned} \tag{7}$$

where TP, TN, FP and FN denote the number of true positives, true negatives, false positives and false negatives. Normally, F1-score and IoU are more concerned by researchers because they take both precision and recall into account and reflect the comprehensive model performance. We evaluated DPIN in all five sample areas and computed the overall values. As shown in Tab. 3, the overall Precision, Recall, F1-score and IoU of DPIN are 90.41%, 92.04%, 83.85% and 91.22%, respectively. DPIN acquired good performances in all sample areas and proved its great ability in the rice mapping tasks. The best performance was acquired in JX1 with an F1-score of 95.64% and IoU of 91.65%, and the worst F1-score of 89.97% was got in ZJ1.

Fig. 6 shows the prediction results of two representative areas in these two areas. Their distributions of ground truth and prediction results are basically identical from a regional perspective, as the purple and blue color areas show in the Figure. Unlike the large-area rectangular plots in the plain of ZJ1, the paddy rice fields of JX1 are distributed along the valleys and slopes and are fragmentized and irregular. Even

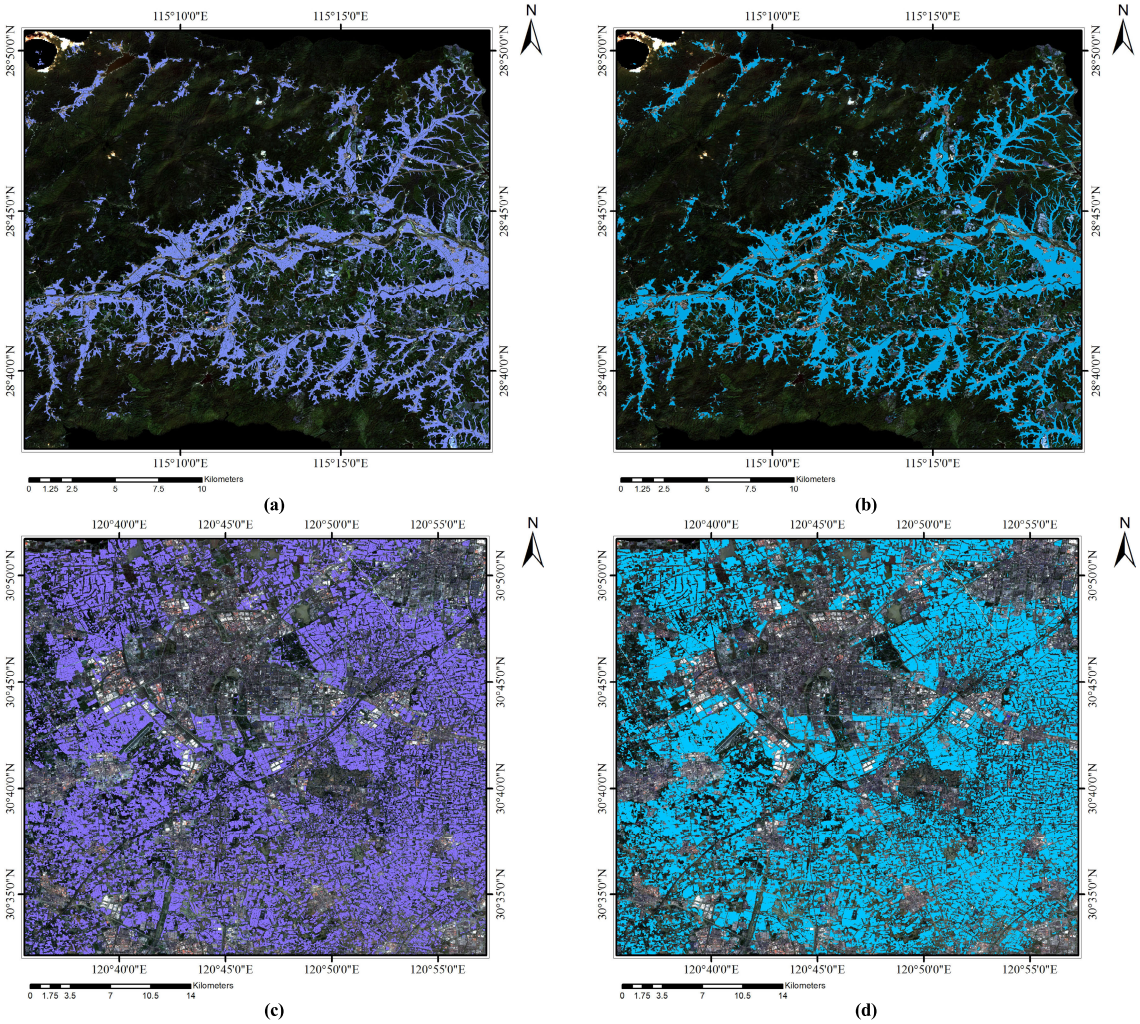

FIGURE 6. Ground truth and model prediction results of two representative areas in JX1 and ZJ1: (a) Ground truth in JX1; (b) Prediction results in JX1; (c) Ground truth in ZJ1; (d) Prediction results in ZJ1. (The backgrounds of (a) and (b) are the integrated Sentinel-2 SR image of August and (c), (d) are the integrated image of October. The backgrounds are chosen for visual presentation only).

so, the best prediction results were still acquired in this area, which is probably due to its simple planting pattern (single rice without rotation). While in most other areas, the rotation patterns vary greatly between different plots and crop types cannot be ascertained, which increases the uncertainties of model learning.

C. PERFORMANCE COMPARISON

DPIN and DPIN-Lite were compared with eight models under the same environment and parameter settings, which include four spatial models and four spatio-temporal models. The spatial models are DeepLab v3+ (with the backbone as MobileNet v2 [50]), Segformer, Mlp-Seg and ConvNeXt, whose structures contain three main components of deep learning: convolution (Deeplab v3+, ConvNeXt), self-attention (Segformer) and fully connected perceptron (Mlp-Seg). The four spatio-temporal models are ConvLSTM, U-ConvLSTM, U-BiConvLSTM and UTAE. All model performances were listed in Tab. 4, and it can be seen that DPIN got the best scores, with an F1-score of 1.09% and IoU of 1.71% higher than the third-best model (UTAE), respectively. DPIN-Lite has fewer parameters and higher inference speed than DPIN and got the second-best performance.

TABLE 4. Comparison of DPIN and peer models on paddy rice mapping.

Model Name	Precision (10^{-2})	Recall (10^{-2})	IoU (10^{-2})	F1-Score (10^{-2})
DeepLab v3+[47]	84.29	88.10	75.68	86.15
Segformer[48]	87.00	90.09	79.40	88.52
Mlp-Seg[49]	87.60	88.52	78.66	88.06
ConvNeXt[44]	90.71	87.57	80.37	89.11
ConvLSTM[34, 35]	87.04	92.23	81.09	89.56
U-ConvLSTM[36]	85.11	93.75	80.54	89.22
U-BiConvLSTM[37]	87.03	92.83	81.55	89.84
UTAE[38]	88.43	91.91	82.04	90.13
DPIN (Ours)	90.41	92.04	**83.85**	**91.22**
DPIN-Lite (Ours)	90.48	91.69	83.62	91.08

Among four spatial models, Deeplab v3+ had the worst performance because its backbone is a lightweight network.

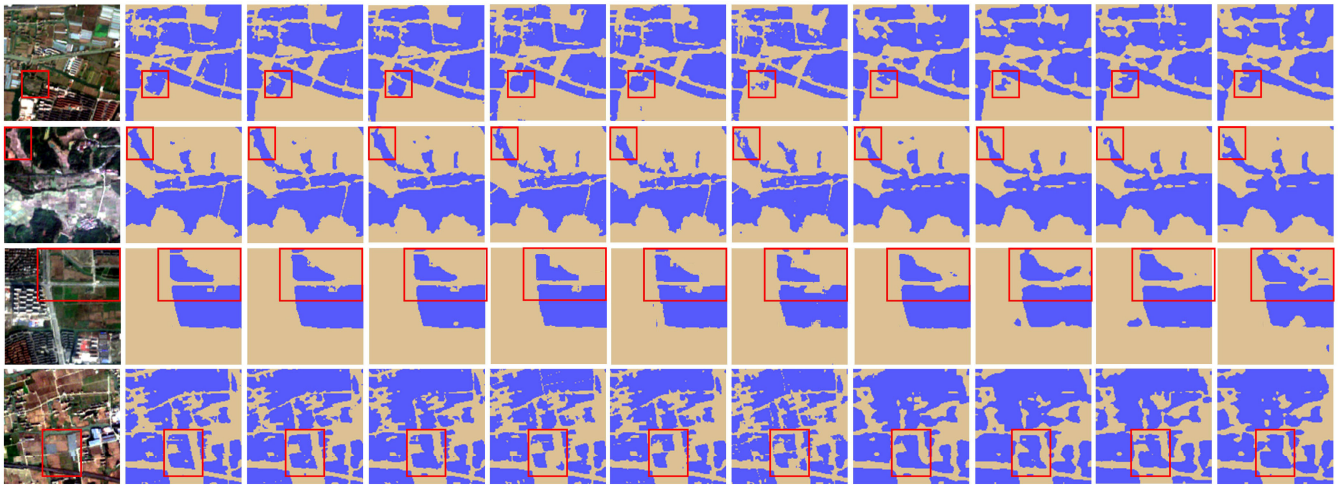

FIGURE 7. Paddy rice mapping results of different models: (a) Images; (b) Ground Truth (GT); (c) DPIN; (d) DPIN-Light; (e) UTAE; (f) U-BiConvLSTM; (g) ConvLSTM; (h) ConvNeXt; (i) MLP-Seg; (j) Segformer; (k) DeepLab v3+. The red boxes highlight the false negatives or false positives. Note that: due to limited spaces, we could not show the prediction results of all the algorithms listed in Table 4.

The other three are constructed by self-attention modules, fully connected perceptron and convolutional blocks, respectively. Results reveal that convolutional structure is more advantageous in paddy rice mapping. However, spatial models generally perform worse than spatio-temporal models because the formers couldn't fully explore and utilize the temporal context relationships.

As a classical spatio-temporal model, the Conv-LSTM model got a not bad F1-score of 89.56%. U-ConvLSTM is constructed by simply adding the LSTM structure as the temporal feature aggregator (TFA) into a U-Net structure. Such modification did not bring better performance because most parameters are still learning spatial features. U-BiConvLSTM and UTAE improved the results (F1-score of 89.84% and 90.13%) by enhancing the capability of TFA. However, the spatial and temporal features still cannot freely interact. DPIN overcomes this problem by creating a dual-path interactive structure and getting the best result.

The performance differences of these models are not only significant in the quantitative assessment but also visually noticeable in the amplifying images as shown in Fig. 7. As indicated in the red rectangle boxes, DPIN captured much finer details compared to other models. To be specific, DPIN produced fewer false positives and negatives (e.g., line 3 & 4 in Fig. 7), returned relatively accurate shapes/boundaries (e.g., line 1, 2 & 4 in Fig. 7) and retained hard-identified plots (e.g., line 2 in Fig. 7).

D. MEMORY COST AND INFERENCE SPEED COMPARISON

The memory cost and inference speed are two main aspects that constrain the spatio-temporal model, and Tab. 5 compares their performances. The memory cost was recorded during the training process, consisting of two parts: the parameters of models themselves, and intermediate variables generated during runtime, termed Activations. The inference speed is measured by Frame-Per-Second (FPS), referring to the throughput of the model per second. The higher the FPS, the faster the model executes.

TABLE 5. Memory cost and inference speed comparison when image size is 128 × 128.

Model Name	Memory cost (M) during training			FPS
	Params	Activations	Total	
ConvLSTM	1299	21746	23045	24.77
U-ConvLSTM	**1283**	**7074**	**8357**	44.57
U-BiConvLSTM	1303	7116	8419	44.45
UTAE	1307	20220	21527	25.75
DPIN	1559	12136	13695	56.21
DPIN-Lite	1395	9400	10795	**74.91**

From Tab. 5 we can see that the main difference in memory cost exists in Activations. U-ConvLSTM and U-BiConvLSTM design relatively simple TFA to save memory and lead to limited performance. Complicated TFA improves the capability of capturing spatio-temporal information at the cost of large memory consumption like UTAE does. DPIN and DPIN-Lite make the appropriate trade-off to get the best performance. And by applying a dual-path interactive branch as a TFA structure instead of time-cost LSTM and self-attention blocks, the execution speed of DPIN and DPIN-Lite are greatly improved. Especially, the speed of DPIN-Lite is about 3 times of the UTAE model (best scores except for DPIN and DPIN-Lite).

E. OBLATION STUDY

To verify the effectiveness of the dual-path interactive structure, a series of comparison models were designed, including 1) DPIN-A: using only one-side interaction from the "separate" path to the "stacked" path; 2) DPIN-B: removal of the "separate" path; 3) DPIN-C: removal of the

TABLE 6. Comparison of DPIN and its variants.

Model	Operations	Precision (10-2)	Recall (10-2)	IoU (10-2)	F1-Score (10-2)
DPIN	-	90.41	92.04	83.85	91.22
DPIN-A	w/o interactive branch from the "stacked" path to the "separate" path	90.47	91.53	83.48	90.99
DPIN-B	w/o the "separate" path	90.39	90.50	82.55	90.44
DPIN-C	w/o the "separate" path and stage 0	89.62	90.57	81.97	90.09

The "w/o" indicates "without".

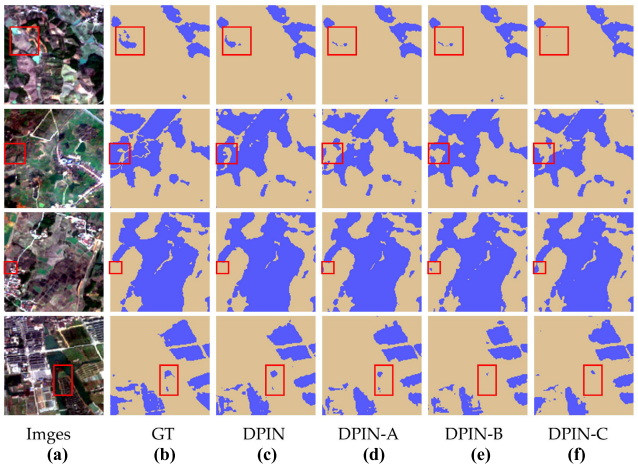

FIGURE 8. Paddy rice mapping results of oblation models: (a) Images; (b) Ground Truth (GT); (c) DPIN; (d) DPIN-A; (e) DPIN-B; (f) DPIN-C. The red boxes highlight false negatives or false positives.

"separate" path and stage 0. The performances of these comparative trials were recorded in Tab. 6, and their scores decreased by the sequence of DPIN, DPIN-A, DPIN-B and DPIN-C.

Through the experiment we can confirm that: First, the dual-path interactive structure could improve the recognition effect. In DPIE, the "separate" path extracts spatial features from sequences by frames, the sequential order retains the temporal context relationship, and the "stacked" path mainly extracts the global spatio-temporal representations. The interaction of the two paths helps the integration of temporal and spatial features. Second, two parallel paths perform better than a single path, because the "separate" path could capture temporal context relationships. Last, the addition of stage 0 without a down-sampling operation improves the results. This demonstrates the importance of the All-level decoder, and with a layer of non-dimensionally sub-sampled features, the fullest spatial detail can be reserved. Some detailed comparisons were visualized in Fig. 8. As the model structure of DPIN is continuously chopped, paddy rice identification is becoming less effective, and the recognition ability for difficult fields gradually decreases. To be specific, the false positives and negatives increased (e.g., lines 1, 3 & 4 in Fig. 7), and the capability of retaining accurate shapes and boundaries decreased (e.g., line 2 in Fig. 7).

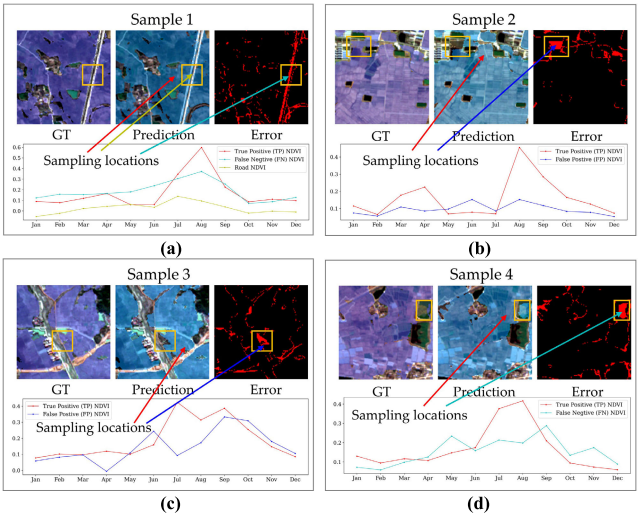

FIGURE 9. Prediction errors of DPIN model. Images with ground truth (GT) label, prediction results and errors are present for 4 typical samples. NDVI curves over a whole year are used to analyze the ground features and causes of mis-prediction. The yellow boxes highlight false negatives or false positives to be analyzed further.

V. DISCUSSION

The superiority of DPIN has been demonstrated above. In this section, limitations and a few methodological attempts of the experiment are discussed. Some prediction errors are shown in Fig. 9, and to analyze the causes and corresponding ground features, NDVI curves of them were plotted. As can be seen, most common errors occur along the plot boundaries. They are mainly caused by the mixed pixels, whose spectral curves consist of several different ground features. Mixed pixels usually affect the classification accuracy of small and linear ground features when using relatively low-resolution images. In Fig. 9(a), the boundary of a road was mis-predicted as its adjacent paddy rice. The NDVI curve of road is flat, the rice curve peaks in August, while the curve trends of their boundary pixels fall in between road and rice, making it difficult to classify. Fig. 9(b) shows a case that the ground truth labels are wrong while our model predicted correctly. Due to the carelessness or other reasons, one block with flat NDVI curve were labeled as paddy rice. It demonstrates that manual labels cannot be 100% right and a well-trained model is able to correct some errors. Fig. 9(c) shows a case where the labels are correct, but the model predicted wrong. This plot has two peaks in the NDVI curves, which is very few in our dataset and then hard to be predicted. It can be deduced that double-season rice is planted here. In Fig. 9(d), the curve of false positive plot has no distinct peaks and valleys, and the crop type and planting pattern are unclear. These ground features belong to the complicated targets of our model.

In this experiment a whole year of 12 images were used, and is it reasonable, or can just using the images in the period of rice growth get a better result? Fig. 10 releases the NDVI curves of five areas (5000 random samples per area). The black curve represents the mean NDVI of paddy rice, and the gray area represents the standard deviation. It can be seen that

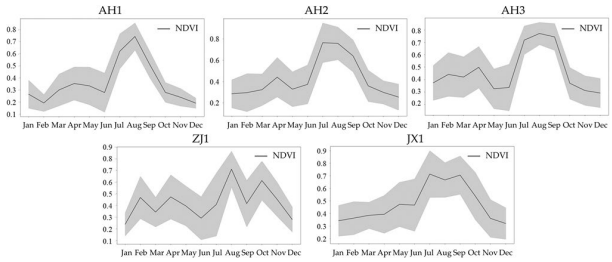

FIGURE 10. NDVI curves and range of paddy rice paddy fields on five areas for 2019 year.

TABLE 7. Mapping performance of DPIN when using different range time-series images.

Model	Time range	Precision (10^{-2})	Recall (10^{-2})	IoU (10^{-2})	F1-Score (10^{-2})
DPIN	Only Aug (1 month)	83.08	89.26	75.52	86.06
	May to Dec (8 month)	88.90	92.32	82.78	90.58
	Jan to Dec (12 month)	90.41	92.04	83.85	91.22

NDVI values in all five areas reach the vertex around August, which means that paddy rice grew to its peak at this time. Also, there is a small peak in April in the curves of AH1, AH2, AH3 and ZJ1, and it's probably the growth peak of the rotated crops. From the curves we can roughly deduce that the single-season paddy rice starts its period from May or June and end in November or December. Thus, time-series images from May to December can cover the whole growth period of paddy rice in our sampling area.

Then a set of comparison experiments were conducted: Rice mapping with a 12-months images (a whole year), eight-months images (May to December), and one-month image (August). Their results are recorded in Tab. 7. We can see that the model performance decreased when using eight-months of images instead of a whole year. This is somewhat counter-intuitive - images outside the paddy rice growth period have no relevant features of paddy rice but adding them into input image series improved the mapping performance. However, it could be explainable from another perspective – images from other months provide some plot-related information: if rotated with other crops, they could reinforce the property of cultivated land; if not rotated, they could provide clear boundaries. Additionally, using a-whole-year time-series images can facilitate the application of proposed model in a large area in spite of the spatial differences of crop phenology and rotation mode. When using the image in August alone, the model performance is worst because the features of other phenological phase are missing.

Trying to improve the mapping results. several more attempts have been explored in the model designing and training processes. The first is adopting stricter data augmentation strategy, including RandAugment [51] and Mixup [52] augmentation. But on the contrary it compromises the mapping accuracy. We speculate that this is because the strict data augmentation operation leads to more significant variation in the sample distribution, and the forced restriction of "label invariance" between the augmented and original samples hurts the model performance. Another attempt is the use of pre-trained technique. DPIN was pre-trained on other datasets and then fine-tuned on the paddy rice-mapping task. The pre-trained data were USDA-NASS Cropland Data Layer (CDL) products. But probably because the growth pattern of U.S.' crops is quite different from that of China, the pre-training did not work as we wished. Note that large-scale pre-training still can be expected to play an important role in the crop mapping tasks in the future.

A few directions deserve further study: One is the trade-off of missing and noisy data, both could decrease the model performance. In this experiment, images with cloud percentage more than 9% were removed. That is to avoid that large pixels were contaminated by clouds, because the pixel-based cloud removal algorithm can hardly detect all clouds, especially thin clouds. If the threshold was set more strictly, more months would have no data. Although our model can learn on incomplete time series images, the performance of the model is expected to be further improved if a more effective balance between cloud presence and image loss can be achieved. Another research direction is fusion of multi-source data. for the cloudy and rainy areas, a fusion input of Sentinel-1 and Sentinel-2 images could be a better choice. In short, DPIN is already a good try in time series images, and further studies may need to focus more on data mining and application on a larger and more complex crop scene.

VI. CONCLUSION

Accurate, large-scale mapping methods for paddy rice fields have long been required by governments and agricultural departments. In this paper, a novel strategy is developed and implemented by using time series Sentinel-2 SR images acquired from the GEE platform. Five sample areas of interest over the middle and lower Yangtze River plain were chosen for method validation. In the strategy, an improved spatio-temporal model DPIN and its lighter version DPIN-Lite, were proposed. Both of them are constructed by a dual-path interactive encoder to enhance the fusion of spatial and temporal features, and an all-level decoder to retain all scale feature maps. Compared with peer models, DPIN yields the best results with an overall F1-score of 91.22%, and has a significant advantage in the inference speed, reaching up to 56.21 FPS when the input image size is 128 × 128. DPIN beats the next-best peer model (UTAE) by 1.09% in F1-score and 118% in inference speed. And DPIN-Lite further improves the inference speed to 291% of the UTAE, with only a 0.14% F1-score decrease from DPIN. Our experiment proves that using a full year of 12 images can get better results. And using full-year images makes our method easily transferred to other areas with no prior crop phenological information. In future, our model will be trained and tested in the larger area with more rotation patterns.

ACKNOWLEDGMENT

The authors would like to thank the data support from the National Earth System Science Data Center,

National Science & Technology Infrastructure of China (http://www.geodata.cn) and also would like to thank the Google Earth Engine Platform for Data Acquisition and Preprocessing.

REFERENCES

[1] C. Kuenzer and K. Knauer, "Remote sensing of rice crop areas," *Int. J. Remote Sens.*, vol. 34, no. 6, pp. 2101–2139, Mar. 2013.
[2] J. Dong, X. Xiao, M. A. Menarguez, G. Zhang, Y. Qin, D. Thau, C. Biradar, and B. Moore, III, "Mapping paddy rice planting area in northeastern Asia with Landsat 8 images, phenology-based algorithm and Google Earth engine," *Remote Sens. Environ.*, vol. 185, pp. 142–154, Mar. 2016.
[3] J. Elliott et al., "Constraints and potentials of future irrigation water availability on agricultural production under climate change," *Proc. Nat. Acad. Sci. USA*, vol. 111, no. 9, pp. 3239–3244, Mar. 2014.
[4] *FAOSTAT*. Accessed: Sep. 9, 2022. [Online]. Available: http://fenix.fao.org/faostat/internal/en/#data/SCL
[5] R. Wassmann, S. V. K. Jagadish, K. Sumfleth, H. Pathak, G. Howell, A. Ismail, R. Serraj, E. Redona, R. K. Singh, and S. Heuer, "Regional vulnerability of climate change impacts on Asian rice production and scope for adaptation," *Adv. Agronomy*, vol. 102, pp. 91–133, Jan. 2009.
[6] D. Sutrisno, W. Ambarwulan, I. Nahib, T. Turmudi, J. Suryanta, R. Windiastuti, and P. Kardono, "Cellular automata Markov method, an approach for rice self-sufficiency projection," *J. Ecol. Eng.*, vol. 20, no. 6, pp. 117–125, Jun. 2019.
[7] H. Jia, F. Chen, C. Zhang, J. Dong, E. Du, and L. Wang, "High emissions could increase the future risk of maize drought in China by 60–70%," *Sci. Total Environ.*, vol. 852, Dec. 2022, Art. no. 158474.
[8] Z. Liu, Z. Li, P. Tang, Z. Li, W. Wu, P. Yang, L. You, and H. Tang, "Change analysis of rice area and production in China during the past three decades," *J. Geograph. Sci.*, vol. 23, no. 6, pp. 1005–1018, Dec. 2013.
[9] Y.-S. Wang, "The challenges and strategies of food security under rapid urbanization in China," *Sustainability*, vol. 11, no. 2, p. 542, Jan. 2019.
[10] X. Xiao, S. Boles, S. Frolking, C. Li, J. Y. Babu, W. Salas, and B. Moore, III, "Mapping paddy rice agriculture in South and Southeast Asia using multi-temporal MODIS images," *Remote Sens. Environ.*, vol. 100, no. 1, pp. 95–113, Jan. 2006.
[11] I. W. Nuarsa, F. Nishio, C. Hongo, and I. G. Mahardika, "Using variance analysis of multitemporal MODIS images for rice field mapping in Bali province, Indonesia," *Int. J. Remote Sens.*, vol. 33, no. 17, pp. 5402–5417, Mar. 2012.
[12] D. Nguyen, K. Clauss, S. Cao, V. Naeimi, C. Kuenzer, and W. Wagner, "Mapping rice seasonality in the Mekong delta with multi-year Envisat ASAR WSM data," *Remote Sens.*, vol. 7, no. 12, pp. 15868–15893, Nov. 2015.
[13] K. Clauss, M. Ottinger, and C. Küenzer, "Mapping rice areas with Sentinel-1 time series and superpixel segmentation," *Int. J. Remote Sens.*, vol. 39, no. 5, pp. 1399–1420, Mar. 2018.
[14] N. Karimi and M. R. Taban, "A convex variational method for super resolution of SAR image with speckle noise," *Signal Process., Image Commun.*, vol. 90, Jan. 2021, Art. no. 116061.
[15] G. S. Klogo, A. Gasonoo, and I. K. E. Ampomah, "On the performance of filters for reduction of speckle noise in SAR images off the coast of the Gulf of Guinea," 2013, *arXiv:1312.2383*.
[16] R. Ni, J. Tian, X. Li, D. Yin, J. Li, H. Gong, J. Zhang, L. Zhu, and D. Wu, "An enhanced pixel-based phenological feature for accurate paddy rice mapping with Sentinel-2 imagery in Google Earth engine," *ISPRS J. Photogramm. Remote Sens.*, vol. 178, pp. 282–296, Aug. 2021.
[17] D. L. Williams, S. Goward, and T. Arvidson, "Landsat," *Photogramm. Eng. Remote Sens.*, vol. 72, no. 10, pp. 1171–1178, Oct. 2006.
[18] R. Torres et al., "GMES Sentinel-1 mission," *Remote Sens. Environ.*, vol. 120, pp. 9–24, May 2012.
[19] M. Drusch, U. D. Bello, S. Carlier, O. Colin, V. Fernandez, F. Gascon, B. Hoersch, C. Isola, P. Laberinti, P. Martimort, A. Meygret, F. Spoto, O. Sy, F. Marchese, and P. Bargellini, "Sentinel-2: ESA's optical high-resolution mission for GMES operational services," *Remote Sens. Environ.*, vol. 120, pp. 25–36, May 2012.
[20] X. Xiao, S. Boles, J. Liu, D. Zhuang, S. Frolking, C. Li, W. Salas, and B. Moore, III, "Mapping paddy rice agriculture in southern China using multi-temporal MODIS images," *Remote Sens. Environ.*, vol. 95, no. 4, pp. 480–492, Apr. 2005.
[21] G. Zhang, X. Xiao, J. Dong, W. Kou, C. Jin, Y. Qin, Y. Zhou, J. Wang, M. A. Menarguez, and C. Biradar, "Mapping paddy rice planting areas through time series analysis of MODIS land surface temperature and vegetation index data," *ISPRS J. Photogramm. Remote Sens.*, vol. 106, pp. 157–171, Aug. 2015.
[22] A. M. Rad, D. Ashourloo, H. S. Shahrabi, and H. Nematollahi, "Developing an automatic phenology-based algorithm for rice detection using Sentinel-2 time-series data," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 12, no. 5, pp. 1471–1481, May 2019.
[23] M. Zhang, H. Lin, G. Wang, H. Sun, and J. Fu, "Mapping paddy rice using a convolutional neural network (CNN) with Landsat 8 datasets in the Dongting lake area, China," *Remote Sens.*, vol. 10, no. 11, p. 1840, Nov. 2018.
[24] H. Li, D. Fu, C. Huang, F. Su, Q. Liu, G. Liu, and S. Wu, "An approach to high-resolution rice paddy mapping using time-series Sentinel-1 SAR data in the Mun river basin, Thailand," *Remote Sens.*, vol. 12, no. 23, p. 3959, Dec. 2020.
[25] S. Asilo, K. D. Bie, A. Skidmore, A. Nelson, M. Barbieri, and A. Maunahan, "Complementarity of two rice mapping approaches: Characterizing strata mapped by hypertemporal MODIS and rice paddy identification using multitemporal SAR," *Remote Sens.*, vol. 6, no. 12, pp. 12789–12814, Dec. 2014.
[26] N. You and J. Dong, "Examining earliest identifiable timing of crops using all available Sentinel 1/2 imagery and Google Earth engine," *ISPRS J. Photogramm. Remote Sens.*, vol. 161, pp. 109–123, Mar. 2020.
[27] M. Boschetti, L. Busetto, G. Manfron, A. Laborte, S. Asilo, S. Pazhanivelan, and A. Nelson, "Phenorice: A method for automatic extraction of spatio-temporal information on rice crops using satellite data time series," *Remote Sens. Environ.*, vol. 194, pp. 347–365, Jun. 2017.
[28] X. Guan, C. Huang, G. Liu, X. Meng, and Q. Liu, "Mapping rice cropping systems in Vietnam using an NDVI-based time-series similarity measurement based on DTW distance," *Remote Sens.*, vol. 8, no. 1, p. 19, Jan. 2016.
[29] L. Tornos, J. A. Domínguez, M. C. Moyano, L. Recuero, V. Cicuéndez, M. J. García-García, and A. Palacios-Orueta, "Assessment of the SASI spectral shape index time series for mapping rice ecosystems in the Mediterranean region," *Agronomy*, vol. 11, no. 7, p. 1365, Jul. 2021.
[30] Q. Yin, M. Liu, J. Cheng, Y. Ke, and X. Chen, "Mapping paddy rice planting area in northeastern China using spatiotemporal data fusion and phenology-based method," *Remote Sens.*, vol. 11, no. 14, p. 1699, Jul. 2019.
[31] S. Zhao, X. Liu, C. Ding, S. Liu, C. Wu, and L. Wu, "Mapping rice paddies in complex landscapes with convolutional neural networks and phenological metrics," *GIScience Remote Sens.*, vol. 57, no. 1, pp. 37–48, Jan. 2020.
[32] W. Zhang, H. Liu, W. Wu, L. Zhan, and J. Wei, "Mapping rice paddy based on machine learning with Sentinel-2 multi-temporal data: Model comparison and transferability," *Remote Sens.*, vol. 12, no. 10, p. 1620, May 2020.
[33] H. C. de Castro Filho, O. A. de Carvalho Júnior, O. L. F. de Carvalho, P. P. de Bem, R. D. S. de Moura, A. O. de Albuquerque, C. R. Silva, P. H. G. Ferreira, R. F. Guimarães, and R. A. T. Gomes, "Rice crop detection using LSTM, bi-LSTM, and machine learning models from Sentinel-1 time series," *Remote Sens.*, vol. 12, no. 16, p. 2655, Aug. 2020.
[34] X. Shi, Z. Chen, H. Wang, D.-Y. Yeung, W.-K. Wong, and W.-C. Woo, "Convolutional LSTM network: A machine learning approach for precipitation nowcasting," in *Proc. NIPS*, vol. 28, Montreal, QC, Canada, 2015, pp. 1–9.
[35] M. Rußwurm and M. Körner, "Convolutional LSTMs for cloud-robust segmentation of remote sensing imagery," 2018, *arXiv:1811.02471*.
[36] R. M. Rustowicz, R. Cheong, L. Wang, S. Ermon, M. Burke, and D. Lobell, "Semantic segmentation of crop type in Africa: A novel dataset and analysis of deep learning methods," in *Proc. CVPR Workshops*, Honolulu, HI, USA, 2019, pp. 75–82.
[37] J. A. C. Martinez, L. E. C. La Rosa, R. Q. Feitosa, I. D. Sanches, and P. N. Happ, "Fully convolutional recurrent networks for multidate crop recognition from multitemporal image sequences," *ISPRS J. Photogramm. Remote Sens.*, vol. 171, pp. 188–201, Jan. 2021.
[38] V. S. Fare Garnot and L. Landrieu, "Panoptic segmentation of satellite image time series with convolutional temporal attention networks," in *Proc. ICCV*, Montreal, QC, Canada, Oct. 2021, pp. 4872–4881.
[39] *National Earth System Science Data Center, National Science & Technology Infrastructure of China*. Accessed: Sep. 9, 2022. [Online]. Available: http://www.geodata.cn

[40] T. Hermosilla, M. A. Wulder, J. C. White, N. C. Coops, G. W. Hobart, and L. B. Campbell, "Mass data processing of time series Landsat imagery: Pixels to data products for forest monitoring," *Int. J. Digit. Earth*, vol. 9, no. 11, pp. 1035–1054, Jun. 2016.
[41] J. Tian, L. Wang, D. Yin, X. Li, C. Diao, H. Gong, C. Shi, M. Menenti, Y. Ge, S. Nie, Y. Ou, X. Song, and X. Liu, "Development of spectral-phenological features for deep learning to understand *Spartina alterniflora* invasion," *Remote Sens. Environ.*, vol. 242, Jun. 2020, Art. no. 111745.
[42] M. Weigand, J. Staab, M. Wurm, and H. Taubenböck, "Spatial and semantic effects of Lucas samples on fully automated land use/land cover classification in high-resolution Sentinel-2 data," *Int. J. Appl. Earth Observ. Geoinf.*, vol. 88, Jun. 2020, Art. no. 102065.
[43] J. L. Ba, J. R. Kiros, and G. E. Hinton, "Layer normalization," *Stat*, vol. 1050, p. 21, Jul. 2016.
[44] Z. Liu, H. Mao, C.-Y. Wu, C. Feichtenhofer, T. Darrell, and S. Xie, "A ConvNet for the 2020s," in *Proc. CVPR*, New Orleans, LA, USA, Jun. 2022, pp. 11976–11986.
[45] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, "Pyramid scene parsing network," in *Proc. CVPR*, Honolulu, HI, USA, 2017, pp. 2881–2890.
[46] T. Xiao, Y. Liu, B. Zhou, Y. Jiang, and J. Sun, "Unified perceptual parsing for scene understanding," in *Proc. ECCV*, Munich, Germany, 2018, pp. 418–434.
[47] L.-C. Chen, G. Papandreou, F. Schroff, and H. Adam, "Rethinking atrous convolution for semantic image segmentation," 2017, *arXiv:1706.05587*.
[48] E. Xie, W. Wang, Z. Yu, A. Anandkumar, J. M. Alvarez, and P. Luo, "SegFormer: Simple and efficient design for semantic segmentation with transformers," in *Proc. NIPS*, vol. 34, 2021, pp. 12077–12090.
[49] I. O. Tolstikhin, N. Houlsby, A. Kolesnikov, L. Beyer, X. Zhai, T. Unterthiner, J. Yung, A. Steiner, D. Keysers, J. Uszkoreit, M. Lucic, and A. Dosovitskiy, "MLP-mixer: An all-MLP architecture for vision," in *Proc. NIPS*, vol. 34, 2021.
[50] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "MobileNetV2: Inverted residuals and linear bottlenecks," in *Proc. CVPR*, Salt Lake City, UT, USA, Jun. 2018, pp. 4510–4520.
[51] E. D. Cubuk, B. Zoph, J. Shlens, and Q. V. Le, "Randaugment: Practical automated data augmentation with a reduced search space," in *Proc. CVPR Workshops*, Jun. 2020, pp. 702–703.
[52] H. Zhang, M. Cisse, Y. N. Dauphin, and D. Lopez-Paz, "mixup: Beyond empirical risk minimization," in *Proc. ICLR*, Vancouver, BC, Canada, 2018, pp. 1–13.

HUI WANG was born in Leping, China, in August 1995. He received the bachelor's and master's degrees in surveying and mapping engineering from the China University of Geosciences, Beijing, China, in 2017 and 2021, respectively. Currently, he is a Junior Researcher with the Nanhu Laboratory. His research interests include intelligent interpretation of remote sensing imagery, design of deep learning network architecture, and application of machine learning model.

BO ZHAO was born in Xianyang, China, in July 1985. He received the bachelor's degree in geographic information system (GIS) from Northwest Agriculture and Forestry University, Xianyang, China, in 2008, and the master's degree in geodetection and information technology and the Doctorate degree in geochemistry from the China University of Geosciences, Beijing, China, in 2011 and 2015, respectively. From 2015 to 2017, he did the postdoctoral research with Chang'an University, Xi'an, China, and Klagenfurt University, Klagenfurt, Austria. He is currently a Senior Research Fellow with the Research Center of Big Data Technology, Nanhu Laboratory. His current research interests include remote sensing, image processing, AI algorithms, and natural geography.

PANPAN TANG was born in Jiyuan, China, in September 1985. He received the bachelor's degree in surveying and mapping engineering from the China University of Geosciences, Beijing, China, in 2008, and the Ph.D. degree in cartography and geographic information system from the Institute of Remote Sensing and Digital Earth, Chinese Academy of Sciences, Beijing, in 2014. From 2014 to 2020, he worked with the Aerospace Information Research Institute, Chinese Academy of Sciences, as an Associate Researcher. He is currently a Senior Research Fellow with the Research Center of Big Data technology, Nanhu Laboratory. His current research interests include deep learning-based remote sensing (synthetic aperture radar-SAR and optical) image processing, e.g., image segmentation, classification, and change detection.

YUXIANG WANG was born in Hunan, China, in September 1975. He received the Ph.D. degree in cartography and geographic information system from the Institute of Remote Sensing and Digital Earth, Chinese Academy of Sciences, Beijing, in 2002. He is currently the Chief Executive Officer of PIESAT Information Technology Company Ltd., and his company developed a serial of remote-sensing related software and platforms named PIE in China.

HAOMING WAN was born in August 1996. He received the bachelor's degree in remote sensing science and technology from the China University of Mining and Technology, Beijing, China, in 2018, and the master's degree in electronics and communication engineering from the University of Chinese Academy of Sciences, Beijing, in 2021. His master's thesis was multi-sensor data fusion for tree species classification of forest stands. He is currently a Research Assistant with the Nanhu Laboratory, Jiaxing, China. His research interest includes artificial intelligence for image processing.

SHI BAI was born in Jinzhong, China, in December 1994. He received the B.S. degree in resource prospecting engineering and the M.S. degree in geological engineering from the China University of Geosciences, Beijing, in 2018 and 2021, respectively. He is currently a Research Assistant with the Research Center of Big Data Technology, Nanhu Laboratory. His current research interests include remote sensing, image processing, AI algorithms, and geological disasters.

RONGHAO WEI received the bachelor's degree from Wuhan University, Beijing, China, in 2005, and the master's degree from the Institute of Geodesy and Geophysics, Chinese Academy of Sciences, Beijing, in 2008. From 2008 to 2020, he worked as an Engineer with the Zhejiang Surveying Institute of Estuary and Coast. He is currently an Engineer with the Zhejiang Institute of Hydraulics and Estuary. His current research interests include remote sensing, image processing, shipboard LiDAR, and sonar data integration.