Deep Learning-Based Spatiotemporal Data Fusion Using a Patch-to-Pixel Mapping Strategy and Model Comparisons

Zurui Ao, Ying Sun, Xiaoyu Pan, and Qinchuan Xin, *Member, IEEE*

***Abstract*— Tradeoffs among the spatial, spectral, and temporal resolutions of satellite sensors make it difficult to acquire remote sensing images at both high spatial and high temporal resolutions from an individual sensor. Studies have developed methods to fuse spatiotemporal data from different satellite sensors, and these methods often assume linear changes in surface reflectance across time and adopt empirical rules and handcrafted features. Here, we propose a dense spatiotemporal fusion (DenseSTF) network based on the convolutional neural network (CNN) to deal with these problems. DenseSTF uses a patch-to-pixel modeling strategy that can provide abundant texture details for each pixel in the target fine image to handle heterogeneous landscapes and models both forward and backward temporal dependencies to account for land cover changes. Moreover, DenseSTF adopts a mapping function with few assumptions and empirical rules, which allows for establishing reliable relationships between the coarse and fine images. We tested DenseSTF in three contrast scenes with different degrees of heterogeneity and temporal changes, and made comparisons with three rule-based fusion approaches and three CNNs. Experimental results indicate that DenseSTF can provide accurate fusion results and outperform the other tested methods, especially when the land cover changes abruptly. The structure of the deep learning networks largely impacts the success of data fusion. Our study developed a novel approach based on CNN using a patch-to-pixel mapping strategy and highlighted the effectiveness of the deep learning networks in the spatiotemporal fusion of the remote sensing data.**

***Index Terms*— Convolutional neural networks (CNNs), deep learning, spatiotemporal fusion.**

I. Introduction

THE fast development of modern satellite technology has advanced the usage of long-term time series of remote sensing images in the monitoring and modeling of the land surface processes [1]–[3]. Given the differences in satellite sensors, orbit altitudes, and revisiting periods, there are tradeoffs regarding the spatial, temporal, and spectral resolutions of images acquired from an individual satellite sensor [4]. For example, the Moderate Resolution Imaging Spectroradiometer (MODIS) provides observations at the spatial resolution ranging from 250 to 1000 m and has a revisit time of nearly one day for most areas across the world. By comparison, Landsat acquires images at a high spatial resolution of 30 m but relatively small scene coverage, and its revisit time is up to 16 days. The recently launched satellite missions (e.g., Sentinel-2) have improved spatial and temporal resolutions compared with Landsat, and they lack historical archives for analysis. In addition, satellite observations are frequently affected by weather conditions, such as clouds and aerosols [5], [6]. As a result, there are large application demands that require continuous remote sensing data at both high spatial and temporal resolutions. Developing spatiotemporal fusion methods to blend remote sensing data acquired by different satellite sensors has, therefore, become a research frontier in the field of remote sensing [7]–[9].

Previous studies have developed different algorithms for the fusion of multisource remote sensing data. Early image fusion algorithms often perform the frequency-domain analysis, such as wavelet decomposition on images, and then fuse the decomposed image layers. For example, Malenovský *et al.* [10] applied wavelet transform to fuse the MODIS and Landsat images, and produced images at MODIS-like temporal resolution and Landsat-like spatial resolution. Zhou *et al.* [11] applied a pyramidal wavelet transform to blend the Landsat and SPOT images. Wu and Wang [12] evaluated the performance of different wavelet functions in blending MODIS and Landsat data, and demonstrated that the fused images generated from these wavelet transformation methods can be used as substitutes when high spatiotemporal resolution images are not available. The multisource image fusion algorithms based on the frequency-domain analysis are simple and fast but often provide fused images with apparent salt-and-pepper noise due to the effects of pixel mixture.

The data fusion approaches based on linear unmixing can effectively deal with the issue of pixel mixture [13]. The linear unmixing approach selects the end-member pixels that have spectral similarities with the target pixel and derives the reflectance of one pixel in the coarse-spatial-resolution image as the linearly weighted averages of the reflectance

Manuscript received April 27, 2021; revised September 14, 2021 and January 23, 2022; accepted February 22, 2022. Date of publication February 24, 2022; date of current version March 31, 2022. This work was supported in part by the National Key Research and Development Program of China under Grant 2017YFA0604300; in part by the Natural Science Foundation of China under Grant U1811464, Grant 41901345, Grant 42171308, and Grant 41875122; in part by the Natural Science Foundation of Guangdong Province under Grant 2021A1515011429; in part by the Western Talents under Grant 2018XBYJRC004; and in part by the Guangdong Top Young Talents under Grant 2017TQ04Z359. *(Corresponding author: Qinchuan Xin.)*

Zurui Ao is with the Faculty of Engineering, Beidou Research Institute, South China Normal University, Guangzhou 510631, China (e-mail: aozurui1990@gmail.com).

Ying Sun and Xiaoyu Pan are with the School of Geography and Planning, Sun Yat-sen University, Guangzhou 510275, China (e-mail: sunying23@mail.sysu.edu.cn; poonsiuyu@126.com).

Qinchuan Xin is with the School of Geography and Planning, Sun Yat-sen University, Guangzhou 510275, China, and also with the State Key Laboratory of Desert and Oasis Ecology, Research Center for Ecology and Environment of Central Asia, Chinese Academy of Sciences, Ürümqi 830011, China (e-mail: xinqinchuan@gmail.com).

Digital Object Identifier 10.1109/TGRS.2022.3154406

of all overlapped pixels in the high-spatial-resolution image. Zhukov *et al.* [14] proposed a multisensor multiresolution technique (MMT) that adopts a moving window to account for the spatiotemporal variability of pixel reflectance. The proposed method is suitable for image fusion applications in regions where the reflectance of adjacent pixels does not vary drastically. Later studies further improved the MMT method and developed models to handle complex landscapes, such as the unmixing-based data fusion (UBDF [15]), the spatial–temporal data fusion approach (STDFA [16]), and the enhanced spatial and temporal data fusion model (ESTDFM [17]). Maselli [18] introduced the distance weights in the moving window and applied larger weights to pixels that are closer to the target pixel. Busetto *et al.* [19] accounted for the spatial and spectral differences among pixels when selecting the end-members and determined the weights of each pixel in the linear unmixing model based on both spatial and spectral similarities. Zurita-Milla *et al.* [20] used the classification approach of ISODATA to extract the class information and improved the fusion of Landsat and MERIS data by introducing constraints on reflectance into the linear unmixing model. The linear unmixing approaches can well explain the relationship of the correspondent pixels between high and low spatial resolution images [21]. However, they normally assume that the spectral reflectance of one feature class is the same across the entire scene, making it difficult for applications in heterogeneous regions. In addition, there are often considerable residuals and outliers in the fusion results when using the linear unmixing models due to the colinearity problem and the noises in the reference image pairs.

To improve the spatiotemporal fusion of multisource remote sensing data and circumvent the problem of residuals outliers in solving the linear unmixing models, Gao *et al.* [22] proposed a spatial and temporal adaptive reflectance fusion model (STARFM), which is widely used in various applications, such as vegetation monitoring and modeling [23]. STARFM searches for pixels that are similar to the central pixel within a moving window and assigns weights according to spatial, spectral, and temporal proximities to reflect the contribution of these similar pixels. Scholars have made improvements based on the STARFM. For example, Hilker *et al.* [24] proposed a spatial–temporal adaptive algorithm for mapping reflectance change (STAARCH) for regions that experience forest disturbance. STAARCH includes a disturbance factor to determine whether the reflectance in Landsat images changes drastically and improves the fusion results by selecting the best phase image in the time series. Zhu *et al.* [4] proposed the enhanced STARFM (ESTARFM) using additional image pairs of MODIS and Landsat. Meng *et al.* [25] developed a spatial and temporal adaptive vegetation index fusion model (STAVFM) that defines a timing window according to the temporal characteristics of crops and improves the temporal weighting strategies by accounting for crop phenology. The virtual image pair-based spatiotemporal fusion (VIPSTF) approach generated virtual image pairs, which is closer to the data on the prediction date in the feature space, to decrease the fusion error caused by temporal changes [26]. These improved methods require at least two pairs of Landsat images and the corresponding MODIS images to identify the changes in the land cover type, while it is difficult to collect sufficient high-quality image pairs within a reasonable period for many applications and for many regions. Zhu *et al.* [27] proposed a flexible spatiotemporal data fusion (FSDAF) method that allows for predicting both gradual change and land cover type change with one reference Landsat-MODIS image pair. The enhanced FSDAF that incorporates subpixel class fraction change information (SFSDAF [28]), FSDAF 2.0 [29], and block-removed spatial unmixing (SU-BR [30]) methods optimized the unmixing process of FSDAF to decrease the fusion residuals. Wang and Atkinson [31] proposed a method that integrates regression model fitting, spatial filtering, and residual compensation (Fit-FC) to deal with strong seasonal changes. The quality of identified similar neighborhoods has a considerable impact on the accuracy of STARFM and its variants. Fu *et al.* [32] improved the searching strategy of pixels by accounting for spectral similarity and land cover distribution in the moving window. Guan *et al.* [33] introduced object-oriented constraints to guide the similar pixel selection. Liu *et al.* [34] extracted the phenological information to decrease the uncertainties resulted from a similar pixel selection procedure. The abovementioned methods can perform the spatiotemporal fusion of multisource remote sensing data effectively [35], [36], but they generally have assumptions on temporal change of surface reflectance and use a series of handcrafted features. The application of these algorithms is challenging in landscapes that have strong heterogeneity and/or experience abrupt changes because the underlying assumptions might not hold in such cases [23], [37], [38].

In recent years, learning-based approaches have gained increasing interest in the field of spatiotemporal fusion. Huang and Song [39] proposed the sparse-representation-based spatiotemporal reflectance fusion model (SPSTFM) that uses the statistical method to learn a relationship between the dictionary pairs of Landsat and MODIS images. Liu *et al.* [40] proposed an extreme learning machine (ELM) that uses a single-layer feedforward neural network to learn the mapping function between fine and coarse images. Machine learning methods, such as regression trees, random forests, and artificial neural networks, have also gained interest in spatiotemporal data fusion [41]–[43]. The fast-developing methods of deep learning provide a new approach for the fusion of remote sensing data. Song *et al.* [44] reconstructed high spatial resolution images from the corresponding low spatial resolution images based on a super-resolution convolutional neural network (SRCNN). Shao *et al.* [45] developed an extended SRCNN for blending Landsat-8 and Sentinel-2 images, and producing frequent and consistent time-series images. Ao *et al.* [46] proposed an attentional SRCNN for blending Landsat-sentinel normalized difference vegetation index (NDVI) and evaluated the influence of method selection and fusion strategy on the fusion accuracy. Note that SRCNN is originally developed for reconstructing super-resolution images with the magnification factor ranging from 2 to 4 [47], while the data fusion of Landsat and MODIS has approximately 16 times magnification factor between image pairs. Following the fusion rules of STARFM, Tan *et al.* [48] proposed a deep convolutional

spatiotemporal fusion network (DCSTFN) to fuse Landsat and MODIS images. Tan *et al.* [49] further improved this algorithm and proposed an enhanced DCSTFN (EDCSTFN) with considerations on the reconstruction errors, feature errors, and structural similarity in the loss function. Liu *et al.* [50] proposed the spatiotemporal fusion network (STFNET) that establishes the relationship between the spectral changes of MODIS and Landsat, and integrates reconstruction error and temporal dependence in the loss function. Both DCSTFN and STFNET assume that the surface reflectance does not change or only changes linearly with time. EDCSTFN requires a threshold to balance different components in the loss function, which is hard to tune in practical applications. STFNET involves user-defined thresholds in both training loss and reconstruction functions. In sum, there is a need to explore and develop new deep learning methods for better blending images acquired from different sensors with few assumptions and empirical rules.

Most of the existing deep learning-based fusion approaches attempt to learn a mapping function between input and output pairs of small image patches (for example, 50 × 50 pixels). The strategy that uses input patches to predict an output patch, hereafter referred to as "the patch-to-patch mapping strategy," might not be robust for use in the deep learning models for the fusion of remote sensing data, especially in landscapes with strong heterogeneity. Compared to natural images that are widely used in the deep learning models, remote sensing images have strong heterogeneity with complex and diverse textures. Due to the effects of spatial autocorrelation and the phenomena of heterogeneity [51], the pixels that are located at the edge of the patches are unlikely to have sufficient spatial structure information for training the patch-to-patch mapping function. By comparison, traditional rule-based fusion algorithms, such as STARFM and its variants, construct a mapping function between input small patches in a moving window and the output pixel in the center of the moving widow. The strategy that uses input patches to predict center pixel, hereafter referred to as "the patch-to-pixel mapping strategy," may be more effective in accounting for spatial heterogeneity, but it has not been tested in the deep learning models for data fusion.

The main objectives of this article are to: 1) develop a novel deep learning-based method with little assumptions and empirical rules; 2) evaluate the effectiveness of patch-to-pixel mapping strategy in spatiotemporal fusion; and 3) evaluate and compare the deep learning-based methods and rule-based fusion algorithms on image fusion.

II. Study Materials

Three contrasting areas named Coleambally Irrigation Area (CIA), Lower Gwydir Catchment (LGC), and Poyang Lake (PYL), respectively, are selected for studies. For each study scene, three pairs of Landsat and MODIS images were used for data fusion. The first and last pairs of images were used for model training, and the second pair of images was used for model prediction. The fused surface reflectance data include the blue (B), green (G), red (R), near-infrared (NIR), shortwave 1 (SW1), and shortwave 2 (SW2) bands. Detailed information regarding the studied scene and data are listed in Table I.

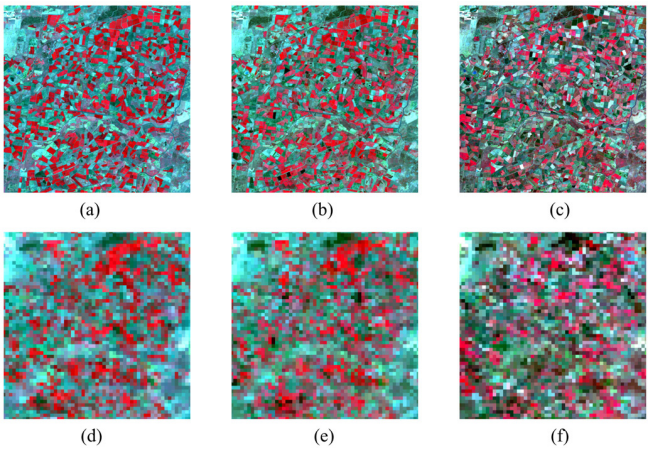

Fig. 1. False-color-composite (NIR, red, and green bands) images for CIA scene with the heterogeneous landscape. The top row shows the Landsat images acquired on (a) 2002/052, (b) 2002/069, and (c) 2002/107, respectively. (d), (e) and (f) in the bottom row are the corresponding MODIS images.

The CIA scene is located in southern New South Wales, Australia (34.003 °S, 145.068 °E) and covers large areas of irrigated croplands with irregular shapes and other land types, such as dryland agriculture and woodlands. Due to the extensive irrigation activities, the surface reflectance changes severely and rapidly in a short time in the irrigated croplands, whereas the surface reflectances in dryland agriculture and woodlands are relatively constant. In Fig. 1, we can see that the CIA scene has strong effects of spatial heterogeneity. The satellite images for the CIA scene came from the benchmark dataset released by Emelyanova *et al.* [52], which were widely used to evaluate the spatiotemporal data fusion methods [27], [31]. The dataset provides coarse and fine resolution image pairs that were radiometrically calibrated and geometrically corrected. The fine resolution images were acquired by Landsat 7 ETM+, and the coarse resolution images were obtained from Terra/MODIS surface reflectance Collection 5 products (MOD09GA). The used Landsat and MODIS images cover an area of 25 km × 25 km comprised of 1000 columns by 1000 lines at 25-m resolution.

The LGC scene with land cover type change is located in northern New South Wales, Australia (29.086 °S, 149.282 °E) and is mainly covered by croplands, bare soil, and natural vegetation. The LGC scene experienced flood events, and the land cover types changed abruptly on the prediction date. A large number of pixels in Fig. 2(a) change to water [see Fig. 2(b)] due to the flooding. The satellite images for the LGC scene came from the benchmark dataset released by Emelyanova *et al.* [52]. The fine resolution images were acquired by Landsat 5 TM, and the coarse resolution images were obtained from MOD09GA Collection 5 products. Data preprocessing, such as radiometric calibration, geocorrection, and atmospheric correction, has been conducted. The used Landsat and MODIS images have a spatial coverage of

TABLE I
INFORMATION ON STUDIED SCENES AND USED IMAGES

Scene	Extent (km^2)	Lat (°)	Lon (°)	Image acquisition date (year/day of year)					
				1st MODIS	1st Landsat	2nd MODIS	2nd Landsat	3rd MODIS	3rd Landsat
CIA	25 × 25	-34.003	145.068	2002/053	2002/052	2002/069	2002/068	2002/108	2002/107
LGC	25 × 25	-29.086	149.282	2004/331	2004/331	2004/347	2004/347	2004/363	2004/363
PYL	30 × 30	28.986	116.628	2017/257	2017/260	2018/099	2018/100	2018/273	2018/277

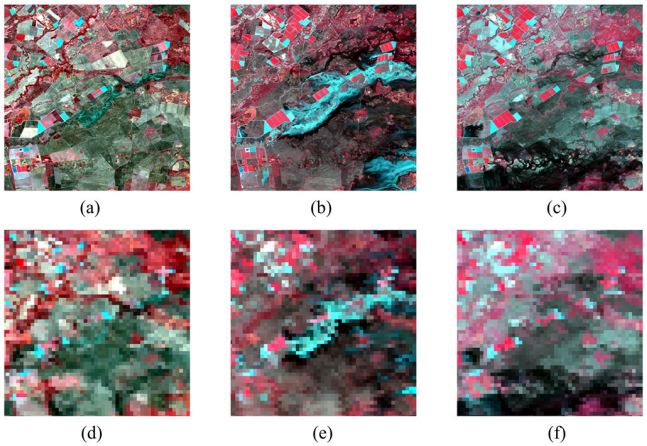

Fig. 2. False-color-composite (NIR, red, and green bands) images for LGC scene with abrupt land cover changes. The top row shows the Landsat images acquired on (a) 2004/331, (b) 2004/347, and (c) 2004/363, respectively. (d), (e) and (f) in the bottom row are the corresponding MODIS images.

25 km × 25 km corresponding to 1000 × 1000 pixels at 25-m resolution.

The PYL scene is located in the south PYL in central China (28.986 °N, 116.628 °E) and indicates strong heterogeneity with various land cover types, such as forest, residential area, wetland, and water body (see Fig. 3). Surface reflectance in this scene is largely influenced by the flood period of the lake and phenological changes of the vegetation, and the reference image pairs are temporally far (approximately 190 days) from the prediction time; thus, we consider it challenging for the data fusion methods. We used the USGS Landsat 8 Tier 1 surface reflectance products and the MOD09GA Collection 6 products as the fine resolution images and the coarse resolution images, respectively. To match the spatial resolution and the coverage of the Landsat 8 images, the corresponding MOD09GAimages were reprojected from the MODIS sinusoidal projection to the Universal Transverse Mercator projection using the HDF-EOS To GeoTIFF Conversion Tool released by the National Aeronautics and Space Administration. The reprojected data were resampled to a 30-m spatial solution using the nearest neighbor interpolation method to preserve the spectral information. To reduce the influence of cloud and cloud shadow in both Landsat and MODIS data, we used the quality control data to generate a mask for pixels with poor qualities and then applied the region-fill algorithm in the commercial software of MATLAB to fill up the surface reflectance of the gap pixels. After the preprocessing, we cropped the images to the size of 1000 × 1000 pixels for training and test.

III. METHODS

The overall goal of developing a spatiotemporal fusion model is to predict the fine image F_1 from the coarse image C_1 on the prediction date t_1 based on two reference pairs of coarse and fine images at two neighboring dates t_0 and t_2 (i.e., images F_0 and C_0 on t_0 and F_2 and C_2 images on t_2, respectively). In essence, there is a need to establish a mapping function Φ between F_1 and the other reference images F_k and C_k

$$F_1 = \Phi(F_k, C_k, C_1). \tag{1}$$

Existing deep learning methods often made assumptions to simplify (1). For example, the deep learning model of DCSTFN [48] uses the mapping function as follows:

$$F_1 = \Phi_1(F_k) + \Phi_2(C_k) - \Phi_3(C_1) \tag{2}$$

where Φ_1, Φ_2, and Φ_3 are three different mapping functions, respectively.

The deep learning model of STFNET assumes that the surface reflectance changes linearly with time [50], and therefore, the underlying mapping function is given as follows:

$$F_1 - F_k = \Phi(C_1 - C_k). \tag{3}$$

Note that the assumptions of (2) and (3) could affect the results of the spatiotemporal fusion of remote sensing images. Given that the convolutional neural network (CNN) has demonstrated powerful capabilities in image classification, feature extraction, and super-resolution [53], [54], we propose a CNN-based data fusion method, namely, the dense spatiotemporal fusion (DenseSTF) network, which establishes the mapping relationship between images at different resolutions directly with little assumptions.

A. DenseSTF

In DenseSTF, we adopt the patch-to-pixel mapping strategy to handle heterogeneous landscapes and model both forward and backward temporal dependencies to account for land cover changes.

The difference between the patch-to-patch and patch-to-pixel mapping strategies is shown in Fig. 4. The patch-to-patch mapping strategy crops the original images into small patches and establishes a mapping function between the input and output patches. In this procedure, the pixel in the center of the output patch can obtain enough texture information, while the pixels located at the edge of the output patch are unlikely to have sufficient texture information due to the missing values [see Fig. 4(a)]. A commonly used method to deal with this

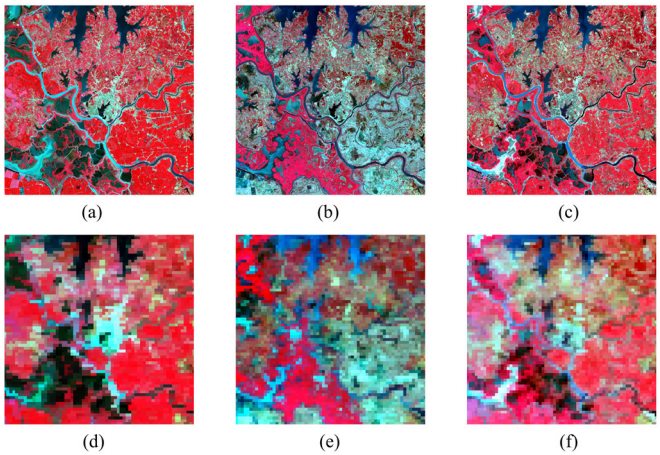

Fig. 3. False-color-composite (NIR, red, and green bands) images for PYL scene with abrupt land cover changes. The top row shows the Landsat images acquired on (a) 2017/260, (b) 2018/100, and (c) 2017/277, respectively. (d), (e) and (f) in the bottom row are the corresponding MODIS images.

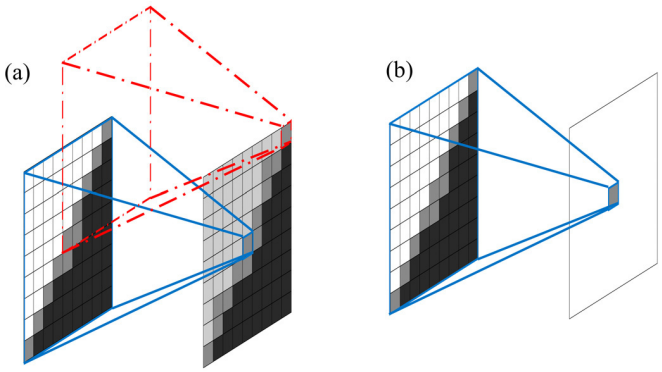

Fig. 4. Diagram shows the difference between (a) patch-to-patch mapping strategy and (b) patch-to-pixel mapping strategy. The patch-to-patch mapping strategy establishes a mapping function between the input and output patches, and the pixels located at the edge of the output patch cannot obtain sufficient texture information due to the missing values. The patch-to-pixel mapping strategy constructs a mapping function between the input small patch in a moving window and the output pixel in the center of the moving widow, which can provide abundant texture information for each pixel.

situation is to fill the missing values with zeros based on the zero padding technique. The filled values inevitably introduce errors to the established mapping function, especially in landscapes with strong heterogeneity. By contrast, the patch-to-pixel mapping strategy uses a moving window to extract input patches from the original images and mapping the relationship between the extracted input patches and the output pixel in the center of the moving widow [see Fig. 4(b)]. This helps to reduce the missing and filled values and ensure that each pixel has sufficient texture information for constructing a reliable mapping function.

We adopt the patch-to-pixel mapping strategy to decrease mapping errors in heterogeneous landscapes. For each pixel in the target fine image F_1, the information from neighboring pixels in a moving window is considered, and then, the mapping function Φ needed to be solved becomes

$$F_1(x_{w/2}, y_{w/2}) = \Phi(F_k(x_i, y_i), C_k(x_i, y_i), C_1(x_i, y_i)) \quad (4)$$

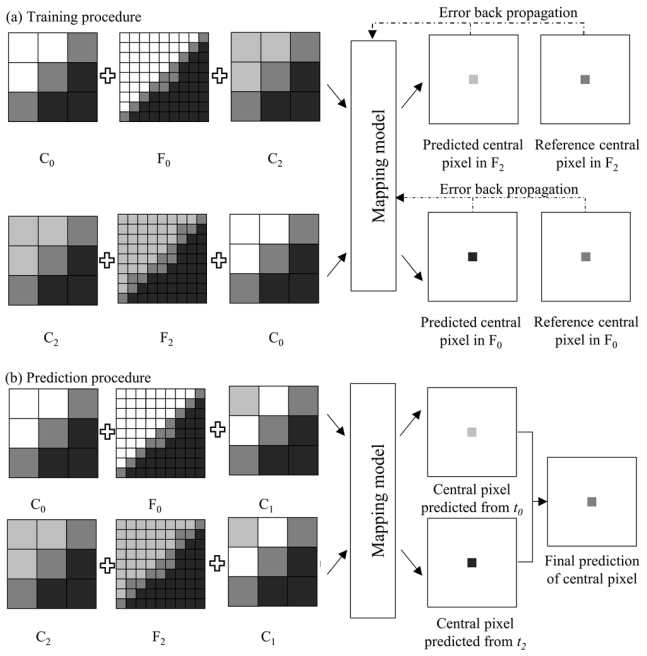

Fig. 5. Framework of the proposed DenseSTF network. The inputs to the model are reference image patches in a moving window, and the outputs of the model are the predicted central pixels in the corresponding moving window.

where w is the moving window size, $(x_{w/2}, y_{w/2})$ denotes the central pixel of the moving window, and (x_i, y_i) represents the ith pixel in the moving window.

A typical deep learning-based method for data fusion consists of: 1) training a CNN model that represents the mapping function Φ using two reference image pairs and 2) applying the trained model to predict a fine image using a coarse image on the prediction date and one pair of the reference image. For example, previous studies normally used $F_0, C_0,$ and C_2 as input training images and F_2 as target training images to learn a CNN model and then adopted the trained model with the inputs of $F_0, C_0,$ and C_1 to predict the image of F_1 [45], [48], [49]. Such a process does not account for the temporal relationships among multitemporal images. We develop a twofold procedure in both model training and prediction (see Fig. 5). Two relationships are constructed in the training procedure of DenseSTF using a shared CNN model Φ with the loss function of mean square errors (mses) to account for the forward temporal dependence from t_0 to t_2 in (5) and the backward temporal dependence from t_2 to t_0 in (6), respectively, as follows:

$$F_2(x_{w/2}, y_{w/2}) = \Phi(F_0(x_i, y_i), C_0(x_i, y_i), C_2(x_i, y_i)) \quad (5)$$
$$F_0(x_{w/2}, y_{w/2}) = \Phi(F_2(x_i, y_i), C_2(x_i, y_i), C_0(x_i, y_i)). \quad (6)$$

In the prediction procedure, DenseSTF makes two initial predictions on date t_1 from the functions that account for the forward and backward temporal dependencies, respectively, as follows:

$$\mathrm{PF}(x_{w/2}, y_{w/2}) = \Phi(F_0(x_i, y_i), C_0(x_i, y_i), C_1(x_i, y_i)) \quad (7)$$
$$\mathrm{PB}(x_{w/2}, y_{w/2}) = \Phi(F_2(x_i, y_i), C_2(x_i, y_i), C_1(x_i, y_i)) \quad (8)$$

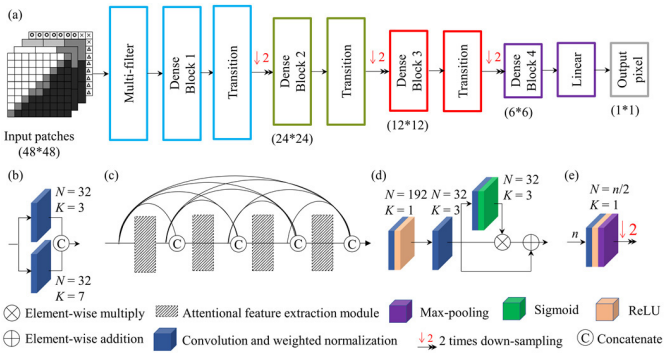

Fig. 6. Structure of the proposed DenseSTF network shows the details of (a) overall network structure, (b) multifilter layer, (c) dense block, (d) attentional feature extraction module in the dense block, and (e) transition layer. N and K denote the number of the feature maps and the kernel size for the convolutional layer, respectively.

where PF denotes the prediction from the function that accounts for forward temporal dependence between t_0 and t_1, and PB denotes the prediction from the function that accounts for the backward temporal dependence between t_2 and t_1.

The final prediction of DenseSTF is concatenated from the two initial predictions as follows:

$$F_1\left(x_{w/2}, y_{w/2}\right) = \frac{\mathrm{PF}\left(x_{w/2}, y_{w/2}\right) + \mathrm{PB}\left(x_{w/2}, y_{w/2}\right)}{2}. \quad (9)$$

There are at least two advantages of the proposed modeling procedures. First, compared with the other deep learning-based methods, the proposed modeling procedure uses a patch-to-pixel modeling strategy that can provide abundant texture details for each pixel in the target fine image to handle heterogeneous landscapes. Different from traditional rule-based methods, the mapping function is built by learning a transformation for all pixels in the moving window rather than weighting similar pixels estimated by handcrafted features. Such a procedure helps preserve abundant spectral and textural information and alleviate the colinearity and ill-posed inversion problems. Second, the proposed modeling procedure accounts for both forward and backward temporal dependencies among multitemporal images to handle the land cover changes. We use a mapping function with few assumptions and empirical rules to model the temporal dependence, which allows for establishing reliable relationships between the coarse and fine images. The process of temporal dependence modeling and the final prediction construction are all based on statistical learning, without involving any changing thresholds. It helps make reliable predictions in various landscapes, particularly in regions with land cover changes.

We develop a new network structure for DenseSTF, as shown in Fig. 6, based on DenseNet [54]. The main idea is to apply the patch-to-pixel mapping strategy to deal with approximately 16 times differences in the spatial resolutions between MODIS and Landsat, and improve the information flow among different convolution layers to achieve accurate predictions. The sizes of input and output images are 48 × 48 (approximately 3 × 3 MODIS pixels) and 1 × 1, respectively.

1) Convolutional Stage: Typical convolutional stage consists of three layers, including: 1) the convolution (CONV) layer to extract image features using a convolution kernel; 2) the batch normalization (BN) layer to solve the internal covariate shift problem in the training process; and 3) the activation function to perform the nonlinear transformation, such as the sigmoid function, the hyperbolic tangent function, and the rectified linear unit (ReLU). The CONV-BN-ReLU structure has been widely used in the semantic segmentation of remote sensing images [53], [54]. In the CONV-BN-ReLU structure, the BN layer uses local statistics (e.g., mean and variance) of each batch for normalization during the training procedure but uses global statistics of all batches for normalization during the prediction procedure. For the application of image fusion, both image patches and batch sizes are relatively small, and there are large differences in local statistics among batches, leading to difficulties in parameter optimization and degraded model performance. DenseSTF uses weight normalization (WN) to accelerate parameter optimization and network convergence as WN does not alter dependencies among examples in a batch [55]. As previous studies have found that removing unnecessary ReLU activations improved the network performance [56] and so does in our preliminary experiments, we remove redundant ReLU activations to reduce model complexity and adopted CONV-WN as the basic convolutional stage to construct the deep learning network.

2) Multifilter Layer: Because low-level features play a key role in the deep learning networks and affect the image fusion results [53], we adopt two convolutional stages with different kernel sizes (i.e., 32*3*3 and 32*7*7 in parallel) to extract multiresolution local features [see Fig. 6(b)]. The multifilter layer delivers sufficient texture and spatial information in varied receptive fields for building robust mapping functions.

3) Dense Block: We modify the feature extraction module in the dense block in DenseNet and develop an attentional feature extraction module to obtain effective representations for the spatial and spectral information. As shown in Fig. 6(d), we first adopt a bottleneck design (i.e., 192*1*1 convolution followed by 32*3*3 convolution) to reduce the complexity of feature extraction and then weight the extracted features via both spatial and channelwise attention maps generated by the attentional module, so as to enhance useful features while suppressing less useful ones. The features obtained from each attentional feature extraction module are concatenated via a series of dense connections to improve the information flow among different modules in the dense block [see Fig. 6(c)]. The modified dense block improves the quality of the extracted features and helps capture useful spatial information in multispectral images.

4) Transition Layer: We use the transition layers to reduce the number of feature maps generated by the dense blocks. In the transition layers, the number of the feature maps is halved, and a pooling layer is used to downsample the spatial resolution of the feature maps. Different from the original DenseNet that uses an average pooling (AP), we apply the maximum pooling to better preserve the boundary information [see Fig. 6(e)].

The number of the bottleneck layers in each dense block was set to 6, 12, 18, and 8, respectively, to balance network performance and computational complexity. The model was implemented in TensorFlow 1.14 and run on an NVIDIA 2080Ti GPU with 11 GB of RAM. The network weights were initialized to 0, and the batch size was set as 128 to fit the GPU memory. We used the Adam optimization method to minimize the training loss. The learning rate was initialized as 5×10^{-4} and decayed with a ratio of 0.7 for every 10 000 iterations. In the training process, the DenseSTF model was iterated for a maximum of 60 000 times to ensure convergence.

B. Comparison Methods

We implemented both deep learning-based spatiotemporal fusion models (i.e., VGG16, STFNET, and EDCSTFN) and rule-based data fusion algorithms (i.e., STARFM, ESTARFM, and FSDAF) for comparison.

The deep learning model of VGG16, initially designated for image classification [57], does not serve the purpose of spatiotemporal data fusion directly. We removed the softmax layer to produce a numeric prediction and modified the number of the output bands for data fusion. VGG16 was implemented using the patch-to-pixel mapping strategy with considerations on temporal dependence to allow for direct comparison with DenseSTF. We initialized weights in VGG16 using random numbers generated by a Gaussian distribution with a mean of zero and a standard deviation of 0.001. The batch size was set as 128 to fit the GPU memory. We used the mse as the loss function during the training and adopted the Adam optimization method to minimize the training loss. The learning rate was initialized as 10^{-4} and halved every 20 000 iterations. In the training process, the VGG16 model was iterated for a maximum of 100 000 times to ensure convergence. The training data used in VGG16 are the same as in DenseSTF. When implementing STFNET, we set 32×32 as the sizes of the input images. The initial output was cropped to produce a smaller output with an image size of 20×20 to avoid the border effects. In the training phase, the training image pairs were cropped with a stride of 20. The training patches were augmented by rotating and reflecting in both vertical and horizontal directions to enrich the training dataset and mitigate model overfitting. After the data augmentation, we obtained 15 000 training patches for each study site. The network weights were initialized to 0, and the batch size was set as 64 to fit the GPU memory. The learning rate was initialized as 10^{-4} and halved every 2000 iterations. We used the Adam optimization method to optimize the parameters and set 100 000 times as the maximum iteration number. When implementing EDCSTFN, we conducted model initialization and optimization according to the guidelines in [49]. The training data used in EDCSTFN were the same as used in STFNET. Note that both DenseSTF and VGG16 adopt the patch-to-pixel mapping strategy for data fusion, and both STFNET and EDCSTFN adopt the patch-to-patch mapping strategy.

FSDAF requires the land cover maps to estimate end-member changes. We classified the study scenes using the iterative self-organizing data analysis (ISODATA) method and interpreted the labels of the obtained land cover classes. Following the guidelines in [27], we set the minimize and maximum numbers of the classes to [3, 5] for all studied scenes. In STARFM, we set the spectral uncertainties of Landsat and MODIS to 0.015 in all studied scenes as there were changes on the land surface across time. STARFM normally produces invalid values in the predicted images; we filled these values with the counterpart of the reference Landsat image to facilitate the accuracy assessment. When implementing ESTARFM, we set the number of classes based on the ISODATA classification maps. The other parameters in the models of STARFM, ESTARFM, and FSDAF were set as default in the published version of the software.

C. Quantitative Assessment

Four metrics, including the correlation coefficient (CC), the root mean square error (RMSE), the average absolute difference (AAD), and the structural similarity index measure (SSIM), are used for quantitative assessment of the models. R is indicative to the linear relationships between the predicted surface reflectance and the reference. RMSE accounts for the errors between the predicted surface reflectance and the reference. AAD accounts for the deviation of the predicted surface reflectance. SSIM assesses the similarity of the overall textures between the predicted images and the reference. These metrics are calculated, respectively, as follows:

$$\mathrm{CC} = \frac{\sigma_{yp}}{\sigma_y * \sigma_p} \tag{10}$$

$$\mathrm{RMSE} = \sqrt{\sum_{i=1}^{n}(y_i - p_i)^2/n} \tag{11}$$

$$\mathrm{AAD} = \sum_{i=1}^{n}|(y_i - p_i)|/n \tag{12}$$

$$\mathrm{SSIM} = \frac{(2\bar{y} * \bar{p} + C_1)(2\sigma_{yp} + C_2)}{(\bar{y}^2 + \bar{p}^2 + C_1)(\sigma_y^2 + \sigma_p^2 + C_2)} \tag{13}$$

where y_i denotes the ith pixel in the reference image y; p_i denotes ith pixel in the predicted image p; $\bar{y}$ and $\bar{p}$ denote the mean of y and p, respectively; σ_y and σ_p denote the variance of y and p, respectively; σ_{yp} denotes the covariance between y and p; n denotes the pixel number; and C_1 and C_2 are constants that ensure the stability of SSIM.

IV. Results

A. Comparison With Rule-Based Methods That Use One Reference Image Pair

Fig. 7 shows the false-color-composite images from the observed data and the spatiotemporal fusion results derived from different methods in the CIA scene. Visual inspection suggests that the fusion image produced by DenseSTF is highly consistent with the observed Landsat image. STFNET tends to produce images with pseudospatial textures, and EDCSTFN appears to generate images with blurred boundaries in croplands. Compared with STFNET and EDCSTFN, the fused image by VGG16 is closer to the observed Landsat

TABLE II
QUANTITATIVE ASSESSMENT OF DIFFERENT FUSION METHODS IN THE CIA SCENE. THE METRIC VALUE INDICATIVE TO THE BEST MODEL PERFORMANCE IS HIGHLIGHTED IN BOLD

Metrics	Bands	DenseSTF	VGG16	STFNET	EDCSTFN	STARFM	FSDAF
CC	B	**0.8749**	0.8570	0.8568	0.8462	0.8620	0.8617
	G	**0.8175**	0.7947	0.7689	0.7304	0.7903	0.7966
	R	**0.8834**	0.8748	0.8617	0.8389	0.8593	0.8661
	NIR	0.9097	0.9072	0.9036	0.8948	0.8664	**0.9139**
	SW1	**0.9004**	0.8948	0.8856	0.8749	0.8700	0.8814
	SW2	**0.9150**	0.9111	0.9085	0.8975	0.8784	0.8954
	Mean	**0.8835**	0.8733	0.8642	0.8471	0.8544	0.8692
RMSE	B	**0.0095**	0.0108	0.0112	0.0134	0.0099	0.0098
	G	**0.0130**	0.0143	0.0148	0.0196	0.0138	0.0136
	R	**0.0196**	0.0202	0.0222	0.0261	0.0208	0.0208
	NIR	0.0410	0.0426	0.0495	0.0416	0.0474	**0.0380**
	SW1	**0.0357**	0.0366	0.0389	0.0412	0.0410	0.0397
	SW2	**0.0324**	0.0332	0.0351	0.0365	0.0389	0.0370
	Mean	**0.0252**	0.0263	0.0286	0.0297	0.0286	0.0265
AAD	B	**0.0071**	0.0083	0.0085	0.0109	0.0072	0.0072
	G	**0.0097**	0.0109	0.0113	0.0151	0.0101	0.0100
	R	**0.0146**	0.0149	0.0170	0.0203	0.0155	0.0155
	NIR	0.0301	0.0313	0.0365	0.0304	0.0323	**0.0278**
	SW1	**0.0245**	0.0249	0.0279	0.0299	0.0288	0.0276
	SW2	**0.0228**	0.0231	0.0260	0.0264	0.0281	0.0266
	Mean	**0.0181**	0.0189	0.0212	0.0222	0.0203	0.0191
SSIM	B	**0.9539**	0.9424	0.9449	0.9100	0.9469	0.9479
	G	**0.9370**	0.9296	0.9285	0.8909	0.9266	0.9297
	R	**0.8932**	0.8855	0.8674	0.8304	0.8705	0.8778
	NIR	0.8168	0.8110	0.7837	0.7899	0.7648	**0.8194**
	SW1	**0.8226**	0.8112	0.7881	0.7874	0.7662	0.7979
	SW2	**0.8317**	0.8209	0.7999	0.8000	0.7607	0.7956
	Mean	**0.8759**	0.8668	0.8521	0.8348	0.8393	0.8614

image. STARFM effectively predicts large objects on the land surface, but the predicted images have considerable spectral differences against the observed Landsat images, and there is apparent salt-and-pepper noise in the croplands. FSDAF largely reduces the effects of the salt-and-pepper noise that is apparent in STARFM.

Quantitative metrics for the CIA scene (see Table II) illustrate that DenseSTF achieves high accuracy. Regarding learning-based algorithms, DenseSTF and VGG16 produce better results than STFNET and EDCSTFN. A possible explanation for this may be related to the superiority of the patch-to-pixel mapping strategy in providing abundant texture information to handle heterogeneous landscapes. When comparing the rule-based methods, FSDAF slightly outperforms STARFM. The possible reason may be the STARFM assumes that the surface reflectance does not change across time, and this assumption is problematic in croplands. Overall, DenseSTF generally produces the best performances for all the metrics in all bands except for the NIR band where FSDAF achieves slightly better performance. When averaged for all bands, the CC, RMSE, AAD, and SSIM values are 0.8835, 0.0252, 0.0181, and 0.8759, respectively, for DenseSTF, and 0.8692, 0.0265, 0.0191, and 0.8614, respectively, for FSDAF. It suggests that the proposed DenseSTF outperforms FSDAF in the CIA scene.

Fig. 8 exhibits the spatiotemporal fusion images derived from different methods for the LGC scene. It can be observed that all methods cannot fully capture the land cover changes as some changes in the Landsat image are invisible in the corresponding MODIS image. STARFM and FSDAF produce the fusion images with unrealistic textures. DenseSTF and VGG16 can better predict the flooded area compared with STFNET and EDCSTFN. There are noticeable MODIS pixel boundaries in the image fused by EDCSTFN. Overall, the fusion result from DenseSTF is the closest to the observed Landsat image.

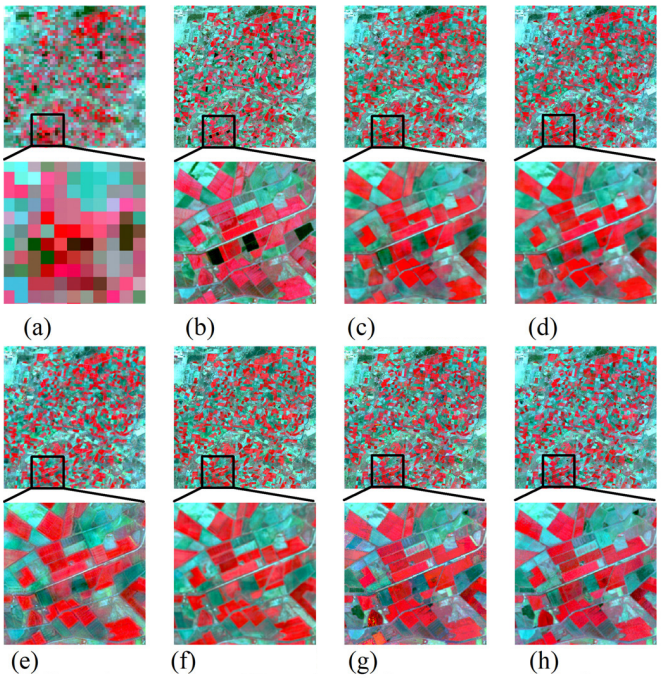

Fig. 7. Results of different spatiotemporal fusion methods for the CIA scene (NIR, red, and green bands as RGB). (a) Observed MODIS image. (b) Observed Landsat image. (c) DenseSTF. (d) VGG16. (e) STFNET. (f) EDCSTFN. (g) STARFM. (h) FSDAF. The lower row images are zoom-in scenes of the area marked in the upper row images. Note that both rule-based methods (STARFM and FSDAF) use only one pair of Landsat and MODIS images as reference images.

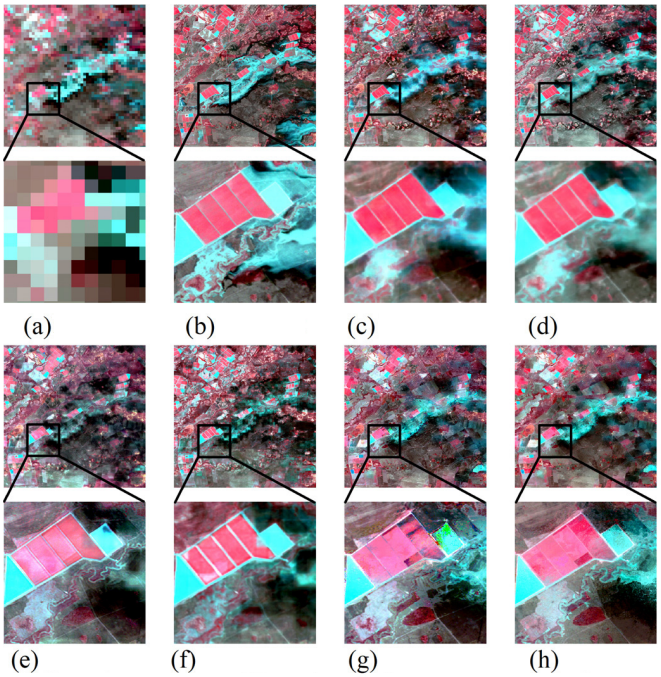

Fig. 8. Same as Fig. 7 but is shown for the LGC scene.

The quantitative analysis for the LGC scene (see Table III) shows that DenseSTF achieves the best accuracy followed by VGG16. The NIR, SW1, and SW2 bands of STARFM results contain a large number of invalid values in the flooded area, indicating that the STARFM is less robust than FSDAF. STFNET and EDCSTFN perform slightly worse than

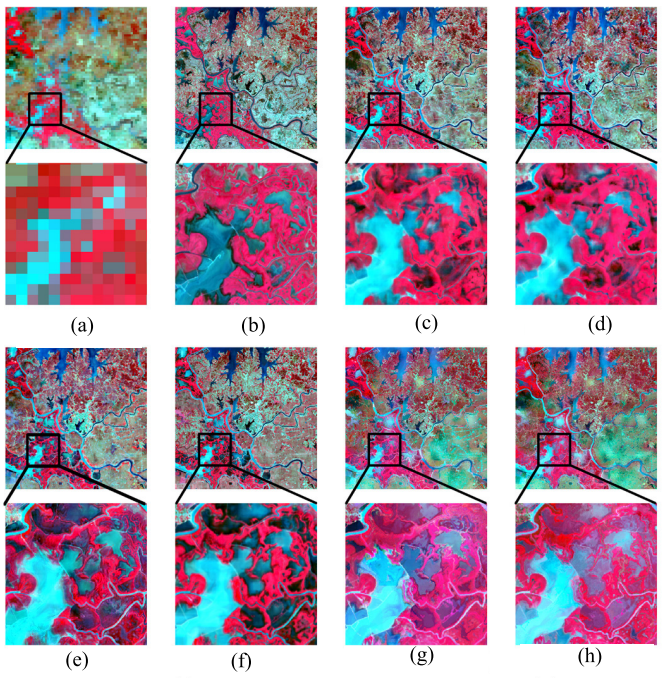

Fig. 9. Same as Fig. 8 but is shown for the PYL scene.

DenseSTF and VGG16. This may have resulted from their underlying assumptions that surface reflectance changes linearly with time. In the flooded area, the land cover changes abruptly, and this assumption can hardly be satisfied, which leads to large uncertainties in the fusion results. By contrast, the proposed modeling procedure accounts for both forward and backward temporal dependencies among multitemporal images and uses a mapping function with few assumptions and empirical rules to model the temporal dependence. Results demonstrate that the proposed modeling procedure is effective in handling the land cover changes.

From the results of the PYL scene (see Fig. 9), it can be observed that the DenseSTF result keeps in line with the observed Landsat image. The enlarged maps for subareas show that DenseSTF well captures the spatial characteristics of the landscapes and makes reasonable predictions on tiny land surface objects. Images fused by STFNET and EDCSTFN contain noticeable artifacts in the enlarged subareas. STARFM effectively models large features, such as water bodies and wide roads on the land surface, but the predicted images have considerable spectral differences against the observed Landsat images. As for the FSDAF result, spatial details have been blurred in the residential region, and pseudospatial textures occur in the enlarged subarea.

As presented in Table IV, DenseSTF yields the best performances for most of the metrics in the PYL scene. The mean CC, RMSE, AAD, and SSIM for all bands are 0.6631, 0.0370, 0.0275, and 0.7449, respectively. Comparison of the metrics in different scenes demonstrates that the predicting power of the spatiotemporal fusion methods decreases as the spatial heterogeneity increases, and the magnitude of land cover changes increases. Moreover, the predicting power of the rule-based spatiotemporal fusion methods decreases faster

TABLE III

QUANTITATIVE ASSESSMENT OF DIFFERENT FUSION METHODS IN THE LGC SCENE. THE METRIC VALUE INDICATIVE TO THE BEST MODEL PERFORMANCE IS HIGHLIGHTED IN BOLD

Metrics	Bands	DenseSTF	VGG16	STFNET	EDCSTFN	STARFM	FSDAF
CC	B	**0.7383**	0.7131	0.6740	0.6010	0.6574	0.6858
	G	**0.7471**	0.7108	0.6555	0.6185	0.6933	0.7137
	R	**0.7321**	0.7119	0.6560	0.6131	0.7059	0.7216
	NIR	0.8259	0.8190	**0.8554**	0.7929	0.8019	0.8229
	SW1	**0.7730**	0.7591	0.7562	0.7494	0.7233	0.7337
	SW2	**0.7649**	0.7482	0.7491	0.7242	0.6959	0.7050
	Mean	**0.7636**	0.7437	0.7244	0.6832	0.7130	0.7305
RMSE	B	**0.0149**	0.0163	0.0170	0.0181	0.0161	0.0156
	G	**0.0207**	0.0229	0.0243	0.0250	0.0223	0.0216
	R	0.0261	0.0278	0.0291	0.0304	0.0260	**0.0254**
	NIR	**0.0347**	0.0354	0.0364	0.0437	0.0373	0.0358
	SW1	**0.0562**	0.0578	0.0613	0.0691	0.0643	0.0645
	SW2	**0.0421**	0.0431	0.0453	0.0563	0.0538	0.0544
	Mean	**0.0325**	0.0339	0.0356	0.0404	0.0366	0.0362
AAD	B	**0.0106**	0.0114	0.0116	0.0132	0.0117	0.0114
	G	**0.0142**	0.0154	0.0156	0.0182	0.0156	0.0151
	R	0.0176	0.0188	0.0185	0.0226	0.0178	**0.0174**
	NIR	**0.0258**	0.0263	0.0254	0.0340	0.0278	0.0269
	SW1	**0.0411**	0.0427	0.0497	0.0529	0.0511	0.0505
	SW2	**0.0306**	0.0319	0.0368	0.0431	0.0427	0.0421
	Mean	**0.0233**	0.0244	0.0263	0.0307	0.0278	0.0272
SSIM	B	**0.9265**	0.9172	0.9245	0.9068	0.9120	0.9171
	G	**0.8974**	0.8868	0.8967	0.8749	0.8792	0.8853
	R	**0.8706**	0.8595	0.8688	0.8424	0.8554	0.8613
	NIR	**0.8125**	0.8038	0.8121	0.7866	0.7820	0.7943
	SW1	**0.6366**	0.6163	0.5782	0.5112	0.5217	0.5441
	SW2	**0.6930**	0.6697	0.6187	0.5630	0.5406	0.5807
	Mean	**0.8061**	0.7922	0.7832	0.7475	0.7485	0.7638

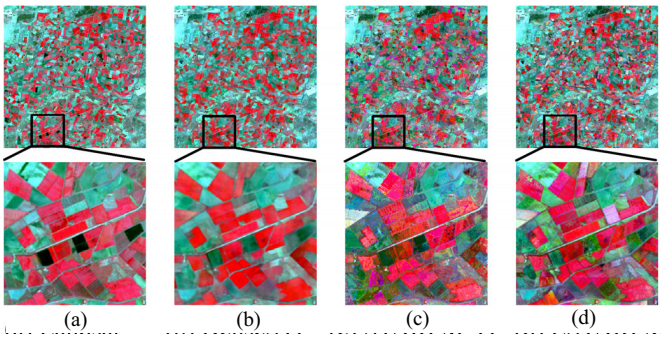

Fig. 10. Comparisons between (a) observed Landsat images and the fusion images derived from (b) DenseSTF, (c) STARFM-TP, and (d) ESTARFM, respectively, in the CIA scene. The suffix of TP stands for the model that uses two pairs of MODIS and Landsat images as inputs.

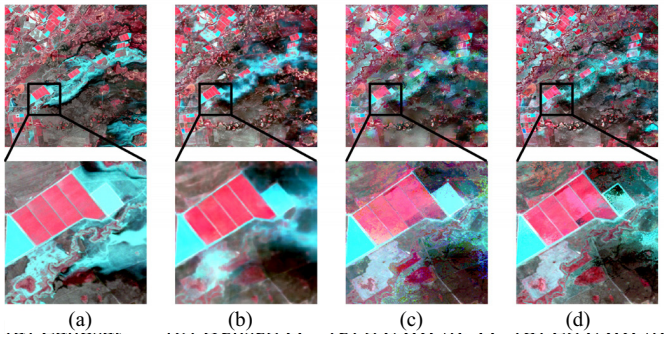

Fig. 11. Same as Fig. 10 but is shown for the LGC scene.

than that of deep learning-based methods. The fusion results obtained from DenseSTF are more robust and reliable than the other methods.

B. Comparison With Rule-Based Methods That Use Two Reference Image Pairs

Figs. 10–12 show the fusion images obtained from different methods using two Landsat-MODIS image pairs as model inputs. As seen from the figure, ESTARFM predicts the object

TABLE IV

QUANTITATIVE ASSESSMENT OF DIFFERENT FUSION METHODS IN THE PYL SCENE. THE METRIC VALUE INDICATIVE TO THE BEST MODEL PERFORMANCE IS HIGHLIGHTED IN BOLD

Metrics	Bands	DenseSTF	VGG16	STFNET	EDCSTFN	STARFM	FSDAF
CC	B	**0.5738**	0.5303	0.5290	0.5279	0.4719	0.4195
	G	**0.4775**	0.4389	0.4080	0.4601	0.3611	0.3009
	R	0.6214	**0.6218**	0.5674	0.5738	0.5595	0.5261
	NIR	**0.7979**	0.7898	0.6642	0.5999	0.4607	0.6411
	SW1	**0.7515**	0.7237	0.7334	0.6909	0.6769	0.6757
	SW2	**0.7566**	0.7156	0.7302	0.7292	0.7263	0.7311
	Mean	**0.6631**	0.6367	0.6054	0.5970	0.5427	0.5491
RMSE	B	**0.0151**	0.0167	0.0184	0.0174	0.0198	0.0205
	G	**0.0230**	0.0242	0.0246	0.0244	0.0276	0.0296
	R	0.0294	**0.0291**	0.0342	0.0327	0.0315	0.0329
	NIR	**0.0608**	0.0621	0.0789	0.0916	0.1175	0.1026
	SW1	**0.0541**	0.0552	0.0562	0.0613	0.0669	0.0711
	SW2	**0.0394**	0.0415	0.0433	0.0426	0.0412	0.0418
	Mean	**0.0370**	0.0381	0.0426	0.0450	0.0508	0.0498
AAD	B	**0.0118**	0.0131	0.0147	0.0137	0.0157	0.0163
	G	**0.0184**	0.0195	0.0195	0.0192	0.0223	0.0245
	R	0.0225	**0.0224**	0.0258	0.0244	0.0245	0.0260
	NIR	**0.0439**	0.0448	0.0610	0.0660	0.0766	0.0850
	SW1	**0.0395**	0.0416	0.0452	0.0454	0.0545	0.0584
	SW2	**0.0290**	0.0310	0.0332	0.0319	0.0319	0.0327
	Mean	**0.0275**	0.0287	0.0332	0.0334	0.0376	0.0405
SSIM	B	**0.9087**	0.8969	0.8957	0.9012	0.8669	0.8517
	G	**0.8676**	0.8578	0.8550	0.8626	0.8240	0.8010
	R	**0.7845**	0.7716	0.7716	0.7651	0.7467	0.7179
	NIR	**0.6159**	0.6113	0.5703	0.5595	0.4290	0.4035
	SW1	**0.6115**	0.5819	0.5586	0.5951	0.5672	0.5534
	SW2	**0.6814**	0.6482	0.6442	0.6652	0.6689	0.6661
	Mean	**0.7449**	0.7280	0.7159	0.7248	0.6838	0.6656

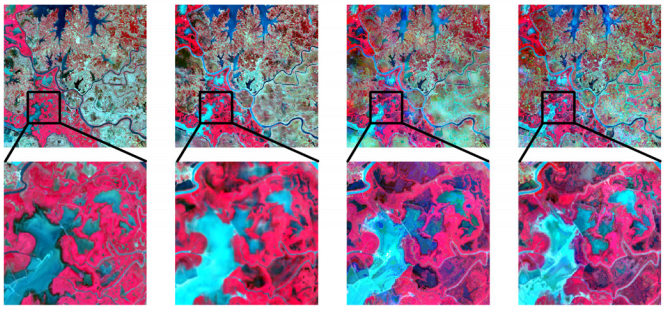

Fig. 12. Same as Fig. 11 but is shown for the PYL scene.

edges well but introduces considerable spectral differences against the observed Landsat images. ESTARFM produces the fusion image with mixed colors within fragmented farmland blocks in the CIA scene. There are considerable effects of salt-and-pepper noise in the images predicted by STARFM-TP in all scenes.

The quantitative analysis (see Table V) shows that DenseSTF outperforms ESTARFM and STARFM-TP. ESTARFM performs better than STARFM-TP in the LGC and PYL scenes, which experiences abrupt land cover changes. Comparing the metrics in Tables II–V, we can find that STARFM-TP outperforms STARFM in the scenes with abrupt changes. ESTARFM yields slightly worse accuracy in the CIA scene. The possible reason may be the surface reflectance changes nonlinearly due to the crop phenology, which is inconsistent with the underlying assumptions of ESTARFM. These findings imply that appropriate consideration of temporal dependence helps contribute to successful data fusion.

C. Comparisons on Models With Different Modules

To understand the impacts of components in the network structure of DenseSTF, we conduct additional modeling

TABLE V

QUANTITATIVE ASSESSMENT OF DIFFERENT METHODS USING TWO LANDSAT-MODIS IMAGE PAIRS. THE METRIC VALUE INDICATIVE TO THE BEST MODEL PERFORMANCE IS HIGHLIGHTED IN BOLD. THE SUFFIX OF TP STANDS FOR THE MODEL THAT USES TWO PAIRS OF MODIS AND LANDSAT IMAGES AS INPUTS

Metrics	Bands	CIA			LGC			PYL		
		DenseSTF	STARFM-TP	ESTARFM	DenseSTF	STARFM-TP	ESTARFM	DenseSTF	STARFM-TP	ESTARFM
CC	B	**0.8749**	0.7268	0.8097	**0.7383**	0.6560	0.6562	**0.5738**	0.5251	0.4717
	G	**0.8175**	0.7170	0.7815	**0.7471**	0.6807	0.6800	**0.4775**	0.4316	0.4491
	R	**0.8834**	0.7830	0.8041	**0.7321**	0.7048	0.6918	**0.6214**	0.5982	0.5708
	NIR	**0.9097**	0.8204	0.8848	**0.8259**	0.8391	0.7916	**0.7979**	0.7437	0.5972
	SW1	**0.9004**	0.8468	0.8641	**0.7730**	0.7547	0.6897	**0.7515**	0.7226	0.7144
	SW2	**0.9150**	0.8708	0.8844	**0.7649**	0.7316	0.5870	**0.7566**	0.7305	0.7213
	Mean	**0.8835**	0.7941	0.8381	**0.7636**	0.7278	0.6827	**0.6631**	0.6253	0.5874
RMSE	B	**0.0095**	0.0144	0.0123	**0.0149**	0.0163	0.0165	**0.0151**	0.0183	0.0204
	G	**0.0130**	0.0153	0.0139	**0.0207**	0.0224	0.0226	**0.0230**	0.0241	0.0260
	R	**0.0196**	0.0254	0.0242	**0.0261**	0.0256	0.0265	**0.0294**	0.0307	0.0337
	NIR	**0.0410**	0.0560	0.0436	**0.0347**	0.0323	0.0367	**0.0608**	0.0695	0.0931
	SW1	**0.0357**	0.0435	0.0416	**0.0562**	0.0575	0.0646	**0.0541**	0.0569	0.0611
	SW2	**0.0324**	0.0395	0.0379	**0.0421**	0.0438	0.0555	**0.0394**	0.0413	0.0418
	Mean	**0.0252**	0.0324	0.0289	**0.0325**	0.0330	0.0371	**0.0370**	0.0401	0.0460
AAD	B	**0.0071**	0.0104	0.0092	**0.0106**	0.0121	0.0124	**0.0118**	0.0145	0.0159
	G	**0.0097**	0.0111	0.0099	**0.0142**	0.0159	0.0163	**0.0184**	0.0191	0.0206
	R	**0.0146**	0.0190	0.0177	**0.0176**	0.0180	0.0191	**0.0225**	0.0235	0.0258
	NIR	**0.0301**	0.0368	0.0307	**0.0258**	0.0237	0.0267	**0.0439**	0.0498	0.0716
	SW1	**0.0245**	0.0297	0.0288	**0.0411**	0.0446	0.0505	**0.0395**	0.0445	0.0488
	SW2	**0.0228**	0.0280	0.0269	**0.0306**	0.0340	0.0433	**0.0290**	0.0318	0.0318
	Mean	**0.0181**	0.0225	0.0205	**0.0233**	0.0247	0.0281	**0.0275**	0.0305	0.0358
SSIM	B	**0.9539**	0.9112	0.9281	**0.9265**	0.9089	0.9102	**0.9087**	0.8750	0.8543
	G	**0.9370**	0.9097	0.9225	**0.8974**	0.8744	0.8781	**0.8676**	0.8420	0.8237
	R	**0.8932**	0.8245	0.8372	**0.8706**	0.8525	0.8555	**0.7845**	0.7535	0.7112
	NIR	**0.8168**	0.7535	0.7754	**0.8125**	0.7978	0.7677	**0.6159**	0.5616	0.4079
	SW1	**0.8226**	0.7492	0.7577	**0.6366**	0.5733	0.5031	**0.6115**	0.5872	0.5858
	SW2	**0.8317**	0.7584	0.7684	**0.6930**	0.6337	0.5092	**0.6814**	0.6638	0.6663
	Mean	**0.8759**	0.8178	0.8316	**0.8061**	0.7734	0.7373	**0.7449**	0.7139	0.6749

studies using different learning architectures, including DenseSTF-BN, which uses BN instead of WN, and DenseSTF-AP, which uses AP instead of maximum pooling. Quantitative model assessments in Table VI suggest that the prediction errors of DenseSTF-AP are slightly higher than DenseSTF. One reason is that AP tends to smooth pixels located on the edges of objects in an image. DenseSTF has higher SSIM indices than DenseSTF-AP, implying that using maximum pooling better preserves texture details. Both DenseSTF and DenseSTF-AP outperform DenseSTF-BN, indicating the advantage of weighted normalization in decreasing the uncertainties caused by the differences in local statistics among batches. Overall, DenseSTF performs the best for all the studied scenes.

D. Spatiotemporal Fusion of Image Time Series

We evaluated the performance of the proposed DenseSTF model in the spatiotemporal fusion of image time series at the CIA and LGC sites, respectively. At the CIA site, nine Landsat-MODIS pairs acquired between 2001/280 and 2002/116 were collected. The first (2004/107) and last (2005/093) image pairs served as the training data, and the other image pairs were used for the model test. For STARFM and FSDAF, the reference Landsat-MODIS pair that was temporally closer to the prediction date was selected as algorithm inputs. Table VII summarizes the fusion accuracies of different methods on each prediction date in the time series. The mean metrics for all bands were shown for comparison. The results

TABLE VI

QUANTITATIVE ASSESSMENT OF THE DENSESTF MODELS USING DIFFERENT MODULES. THE METRIC VALUE INDICATIVE TO THE BEST MODEL PERFORMANCE IS HIGHLIGHTED IN BOLD. DENSESTF-BN USES BN INSTEAD OF WEIGHTED NORMALIZATION, AND DENSESTF-AP USES AP INSTEAD OF MAXIMUM POOLING

Metrics	Bands	CIA			LGC			PYL		
		DenseSTF	DenseSTF-AP	DenseSTF-BN	DenseSTF	DenseSTF-AP	DenseSTF-BN	DenseSTF	DenseSTF-AP	DenseSTF-BN
CC	B	**0.8749**	0.8694	0.8651	**0.7383**	0.7283	0.6942	**0.5738**	0.5716	0.5389
	G	0.8175	**0.8183**	0.8111	**0.7471**	0.7396	0.7133	**0.4775**	0.4630	0.4419
	R	0.8834	**0.8847**	0.8784	**0.7321**	0.7290	0.6940	0.6214	0.6128	**0.6240**
	NIR	**0.9097**	0.9048	0.9016	**0.8259**	0.8189	0.8211	**0.7979**	0.7846	0.7930
	SW1	**0.9004**	0.9003	0.8997	**0.7730**	0.7685	0.7642	**0.7515**	0.7314	0.7458
	SW2	**0.9150**	0.9142	0.9147	**0.7649**	0.7639	0.7646	**0.7566**	0.7518	0.7511
	Mean	**0.8835**	0.8820	0.8784	**0.7636**	0.7580	0.7419	**0.6631**	0.6525	0.6491
RMSE	B	**0.0095**	0.0099	0.0102	**0.0149**	0.0157	0.0167	**0.0151**	0.0153	0.0171
	G	**0.0130**	0.0131	0.0133	**0.0207**	0.0219	0.0222	**0.0230**	0.0236	0.0242
	R	**0.0196**	0.0197	0.0200	**0.0261**	0.0275	0.0284	**0.0294**	0.0295	0.0299
	NIR	**0.0410**	0.0423	0.0434	**0.0347**	0.0349	0.0357	**0.0608**	0.0649	0.0617
	SW1	**0.0357**	0.0359	0.0356	0.0562	0.0568	**0.0551**	0.0541	0.0579	**0.0526**
	SW2	**0.0324**	0.0325	0.0325	**0.0421**	0.0423	0.0428	**0.0394**	0.0408	0.0395
	Mean	**0.0252**	0.0256	0.0258	**0.0325**	0.0332	0.0335	**0.0370**	0.0387	0.0375
AAD	B	**0.0071**	0.0074	0.0076	**0.0106**	0.0111	0.0117	**0.0118**	0.0122	0.0133
	G	**0.0097**	0.0099	0.0100	**0.0142**	0.0147	0.0151	**0.0184**	0.0185	0.0194
	R	**0.0146**	0.0147	0.0149	**0.0176**	0.0183	0.0191	**0.0225**	0.0229	0.0231
	NIR	**0.0301**	0.0309	0.0321	**0.0258**	0.0259	0.0269	**0.0439**	0.0470	0.0446
	SW1	**0.0245**	0.0246	0.0245	0.0411	0.0417	**0.0409**	**0.0395**	0.0413	0.0397
	SW2	**0.0228**	0.0229	0.0229	**0.0306**	0.0309	0.0312	**0.0290**	0.0310	0.0298
	Mean	**0.0181**	0.0184	0.0187	**0.0233**	0.0238	0.0242	**0.0275**	0.0288	0.0283
SSIM	B	**0.9539**	0.9508	0.9491	**0.9265**	0.9155	0.9191	**0.9087**	0.9075	0.9017
	G	**0.9370**	0.9369	0.9341	**0.8974**	0.8885	0.8921	**0.8676**	0.8669	0.8654
	R	**0.8932**	0.8924	0.8878	**0.8706**	0.8632	0.8605	**0.7845**	0.7844	0.7829
	NIR	**0.8168**	0.8127	0.8055	**0.8125**	0.8119	0.8023	**0.6159**	0.6089	0.6206
	SW1	**0.8226**	0.8199	0.8182	0.6366	0.6330	**0.6371**	**0.6115**	0.6109	0.5986
	SW2	**0.8317**	0.8276	0.8276	**0.6930**	0.6908	0.6912	**0.6814**	0.6651	0.6680
	Mean	**0.8759**	0.8734	0.8704	**0.8061**	0.8005	0.8004	**0.7449**	0.7406	0.7395

(see Table VII) showed that the RMSE values were lower on all dates for DenseSTF compared with the other methods. The accuracy of fusion results tended to decrease as the temporal distance from the reference date increases. Overall, DenseSTF produced more accurate results than the other tested methods.

At the LGC site, we used the Landsat-MODIS pairs captured on 2004/107 and 2005/093 for model training and fused image time series on ten prediction dates (see Table VIII). It can be observed from Table VIII that DenseSTF was reasonably accurate. The metrics of both RMSE and AAD were lower, and the metrics of both CC and SSIM were higher than the corresponding metrics for the other methods. The result obtained for the LGC site is in line with that obtained for the CIA site.

V. DISCUSSION

This article proposes a new deep learning-based model that adopts a patch-to-pixel mapping strategy to perform a spatiotemporal fusion of Landsat and MODIS data. The proposed deep learning model of DenseSTF produces better fusion images than the other tested methods, demonstrating the applicability of the deep learning methods in spatiotemporal data fusion. Although DenseSTF produces accurate fusion images in general, it does not guarantee that DenseSTF achieves the best assessment metrics than the other methods in all circumstances. Note that we used mse for all spectral bands as the loss function when training both VGG16 and DenseSTF. The method that minimizes overall mse does not necessarily optimize the model performance for each spectral band, and optimizing the loss of mse does not always lead

TABLE VII
COMPARISON OF DIFFERENT METHODS FOR TIME SERIES PREDICTION IN CIA SCENE. THE METRIC VALUES INDICATIVE TO THE BEST MODEL PERFORMANCE ARE HIGHLIGHTED IN BOLD

Date (year/day of year)	Metrics	DenseSTF	VGG16	STFNET	EDCSTFN	STARFM	FSDAF	ESTARFM
2001/289	CC	**0.8097**	0.7940	0.7275	0.6875	0.7134	0.8156	0.7533
	RMSE	**0.0346**	0.0355	0.0427	0.0682	0.0480	0.0348	0.0403
	AAD	0.0244	0.0248	0.0321	0.0499	0.0261	**0.0240**	0.0277
	SSIM	**0.8934**	0.8863	0.8514	0.7984	0.8822	0.8914	0.8626
2001/305	CC	**0.7463**	0.7230	0.6825	0.6881	0.6057	0.7073	0.6469
	RMSE	**0.0396**	0.0417	0.0465	0.0708	0.0558	0.0435	0.0480
	AAD	**0.0292**	0.0305	0.0357	0.0522	0.0338	0.0316	0.0347
	SSIM	**0.8625**	0.8497	0.8284	0.7854	0.8349	0.8444	0.8164
2001/337	CC	0.5366	**0.5505**	0.5175	0.4519	0.4040	0.4940	0.5006
	RMSE	**0.0510**	0.0519	0.0559	0.0867	0.0606	0.0556	0.0569
	AAD	**0.0393**	0.0402	0.0461	0.0671	0.0434	0.0419	0.0427
	SSIM	**0.8252**	0.8186	0.8135	0.7285	0.7908	0.8061	0.7790
2002/004	CC	**0.5156**	0.5008	0.4467	0.1947	0.4023	0.4767	0.5127
	RMSE	**0.0647**	0.0652	0.0700	0.1199	0.0675	0.0652	0.0664
	AAD	**0.0501**	0.0502	0.0593	0.0981	0.0511	0.0502	0.0502
	SSIM	**0.7763**	0.7759	0.7653	0.6364	0.7477	0.7639	0.7406
2002/052	CC	**0.5546**	0.5309	0.4132	0.4830	0.3218	0.3603	0.5205
	RMSE	**0.0491**	0.0500	0.0537	0.0829	0.0612	0.0588	0.0528
	AAD	**0.0374**	0.0383	0.0427	0.0638	0.0449	0.0457	0.0393
	SSIM	**0.8214**	0.8150	0.8028	0.7365	0.7618	0.7680	0.7883
2002/077	CC	0.5979	0.5807	0.5044	0.5147	0.5956	**0.5985**	0.5736
	RMSE	**0.0434**	0.0444	0.0474	0.0743	0.0455	0.0446	0.0480
	AAD	**0.0325**	0.0335	0.0370	0.0563	0.0342	0.0337	0.0351
	SSIM	**0.8532**	0.847	0.8381	0.7632	0.8524	0.8530	0.8207
2002/107	CC	**0.7602**	0.7529	0.6989	0.6701	0.7463	0.7599	0.7338
	RMSE	**0.0278**	0.0279	0.0319	0.0521	0.0288	0.0284	0.0311
	AAD	**0.0207**	0.0208	0.0249	0.0394	0.0177	0.0172	0.0218
	SSIM	**0.9189**	0.9160	0.8965	0.8337	0.9067	0.9121	0.9032

to the optimized values in the other quantitative metrics, such as CC, AAD, and SSIM. There might be tradeoffs among the accuracy of different bands during the model training. One possible solution to further improve the model performance is to train an independent network for each spectral band, but this strategy largely increases computational complexity. Nevertheless, a comparison of overall fusion accuracies of all bands between different methods (see Tables II–VI) demonstrates that the mean CC, RMSE, AAD, and SSIM of DenseSTF are superior to those of the other methods. Overall, DenseSTF achieves reasonable performance on producing the fusion images using Landsat and MODIS image pairs across different scenes.

Among the tested deep learning-based models, both DenseSTF and VGG16 perform substantially better than STFNET and EDCSTFN (see Tables II–VI) in all scenes. One possible explanation is both STFNET and EDCSTFN use the patch-to-patch mapping strategies for model training and image fusion, and the patch-to-patch mapping strategy is unlikely to deliver sufficient spatial structure and texture information for pixels at the edge of an image patch, particularly in heterogeneous regions. Another possible reason may be that the underlying assumptions on linear changes of surface reflectance cannot be satisfied in the tested scenes. In addition, the loss functions of both STFNET and EDCSTFN consist of different components, such as spectral losses, feature losses, textural losses, and temporal losses. Model tuning is required to balance these components, but it is not easy to find the best parameter settings in heterogeneous landscapes and in regions with land cover changes. By contrast, DenseSTF uses the patch-to-pixel mapping strategy to ensure sufficient spatial structure and texture information for network modeling and account for

TABLE VIII

COMPARISON OF DIFFERENT METHODS FOR TIME SERIES PREDICTION IN LGC SCENE. THE METRIC VALUES INDICATIVE TO THE BEST MODEL PERFORMANCE ARE HIGHLIGHTED IN BOLD

Date (year/day of year)	Metrics	DenseSTF	VGG16	STFNET	EDCSTFN	STARFM	FSDAF	ESTARFM
2004/123	CC	**0.9013**	0.8929	0.8805	0.8693	0.7907	0.9089	0.8854
	RMSE	**0.0217**	0.0224	0.0258	0.0420	0.0344	0.0223	0.0227
	AAD	**0.0162**	0.0169	0.0203	0.0320	0.0164	0.0164	0.0163
	SSIM	**0.9529**	0.9485	0.9375	0.8931	0.9461	0.9513	0.9447
2004/187	CC	**0.7936**	0.7569	0.7543	0.6653	0.3940	0.7036	0.7138
	RMSE	**0.0307**	0.0332	0.0346	0.0682	0.0615	0.0342	0.0361
	AAD	**0.0229**	0.0251	0.0272	0.0517	0.0277	0.0254	0.0260
	SSIM	**0.9190**	0.9068	0.9039	0.8257	0.8791	0.8874	0.8731
2004/219	CC	**0.7168**	0.6808	0.7080	0.5666	0.3185	0.5711	0.5328
	RMSE	**0.0481**	0.0520	0.0519	0.1005	0.0792	0.0508	0.0581
	AAD	**0.0383**	0.0424	0.0433	0.0807	0.0405	0.0379	0.0422
	SSIM	**0.8066**	0.7818	0.7826	0.6819	0.7725	0.7837	0.7459
2004/235	CC	**0.7985**	0.7503	0.7647	0.6915	0.3876	0.7030	0.6695
	RMSE	**0.0375**	0.0414	0.0429	0.0777	0.0700	0.0392	0.0454
	AAD	**0.0290**	0.0328	0.0351	0.0614	0.0328	0.0296	0.0330
	SSIM	**0.8850**	0.8625	0.8598	0.7750	0.8396	0.8532	0.8177
2004/299	CC	**0.7459**	0.7020	0.6939	0.7080	0.6029	0.6555	0.7251
	RMSE	**0.0267**	0.0293	0.0277	0.0472	0.0324	0.0298	0.0293
	AAD	**0.0220**	0.0240	0.0227	0.0383	0.0211	0.0229	0.0228
	SSIM	**0.9576**	0.9511	0.9562	0.9078	0.9259	0.9277	0.9299
2004/331	CC	**0.7578**	0.7299	0.7273	0.7261	0.6181	0.6397	0.6944
	RMSE	**0.0302**	0.0331	0.0337	0.0518	0.0363	0.0371	0.0365
	AAD	**0.0236**	0.0267	0.0270	0.0410	0.0266	0.0290	0.0287
	SSIM	**0.9495**	0.9442	0.9453	0.9041	0.9160	0.9184	0.9177
2005/013	CC	**0.8505**	0.8243	0.7763	0.8112	0.7517	0.8257	0.8094
	RMSE	**0.0271**	0.0288	0.0329	0.0514	0.0344	0.0298	0.0310
	AAD	**0.0204**	0.0219	0.0243	0.0382	0.0228	0.0232	0.0234
	SSIM	**0.9491**	0.9425	0.9358	0.8920	0.9314	0.9327	0.9270
2005/024	CC	**0.8565**	0.8413	0.7865	0.7979	0.7801	0.8432	0.8251
	RMSE	**0.0296**	0.0308	0.0384	0.0591	0.0365	0.0301	0.0320
	AAD	**0.0211**	0.0222	0.0282	0.0416	0.0220	0.0219	0.0226
	SSIM	**0.9325**	0.9260	0.9134	0.8612	0.9173	0.9148	0.9066
2005/061	CC	**0.8818**	0.8578	0.8157	0.8332	0.8213	0.8651	0.8428
	RMSE	**0.0244**	0.0260	0.0303	0.0470	0.0292	0.0267	0.0284
	AAD	**0.0180**	0.0191	0.0214	0.0330	0.0195	0.0201	0.0206
	SSIM	**0.9562**	0.9491	0.9431	0.9026	0.9438	0.9443	0.9358
2005/093	CC	**0.9029**	0.8846	0.8402	0.8490	0.8362	0.9019	0.8801
	RMSE	**0.0206**	0.0222	0.0262	0.0413	0.0272	0.0211	0.0230
	AAD	**0.0143**	0.0156	0.0175	0.0270	0.0149	0.0152	0.0158
	SSIM	**0.9688**	0.9623	0.9569	0.9207	0.9639	0.9638	0.9567

both spatial and temporal dependencies among image pairs. The mapping function generated by DenseSTF requires little assumptions or empirical settings and, hence, is more reliable in different circumstances.

Previous studies have found that CNNs often suffer from the data-hungry problem [58], and therefore, increasing the number of reference image pairs likely helps improve the fusion accuracy [45]. When using multitemporal images, it is necessary to account for temporal dependence among images during model training but remains challenging. One commonly used method is to assign weights to images according to the time intervals with the predicted image [22]. This method could be problematic in areas where surface reflectance changes nonlinearly with time [4]. Studies also attempted to use the spectral differences between coarse-spatial-resolution images to measure the degrees of temporal changes [4], [50]. Due to the effects of pixel mixture, this approach may not be feasible in heterogeneous landscapes [23]. We apply a straightforward strategy in DenseSTF to account for both forward and backward temporal dependencies among images. Our experimental results support the effectiveness of the adopted strategy across landscapes. Nevertheless, to allow for comparisons with the other fusion methods, we only tested two pairs of reference images in DenseSTF, and further investigation is needed for understanding its performance in spatiotemporal fusion with multiply pairs of reference images.

Similar to the other spatiotemporal fusion methods, the proposed DenseSTF is suitable for blending the surface reflectance acquired by other satellite sensors, such as Sentinel-2 and Sentinel-3 images [31]. Apart from surface reflectance, it is also of interest to fuse the products derived from surface reflectances, such as NDVI [59], the leaf area index [60], and the land surface temperature [35]. Theoretically, traditional rule-based methods require the fused products to be linearly additive and, hence, inevitably introduce errors when fusing nonlinearly additive products. By comparison, the deep learning-based methods can directly perform nonlinear transformation via hidden layers, and the errors can be largely reduced [46]. Therefore, DenseSTF may be a better choice in applications where the fusion of nonlinearly additive products is needed.

Comparisons among models using different modules imply that the network structure is important to accelerate model convergence and improve the model performance. In recent years, a number of novel network structures (e.g., batch renormalization, the Gaussian error linear unit, and the recurrent neural network) have proved effective in image semantic segmentation [61]–[64]. Testing of the other network structures in DenseSTF is worthy of studying in near future for improving the model applicability.

For spatiotemporal fusion methods, it is generally difficult to predict abrupt changes. For example, the black blocks in Figs. 7 and 8 are not restored by all methods. One major challenge is that the acquisition time of the fine-resolution image and the coarse-resolution image could be different, and the abrupt changes (e.g., irrigation and flood events) could bring noticeable spectral inconsistencies between the fine-resolution and coarse-resolution images. In such cases, the changes in the fine-resolution image are invisible in the corresponding coarse-resolution image. As a result, all methods cannot fully capture the abrupt changes. A possible solution to this problem is to use more auxiliary data (e.g., Sentinel-2 images) that are temporally close to the prediction time to provide sufficient information for the spectral changes.

Although deep learning-based methods can yield higher fusion accuracy, they normally require more computing resources than traditional rule-based methods. Fortunately, the emergence of cloud computing platforms, such as Google Earth Engine (GEE) [65] and the Multi-Mission Algorithm and Analysis Platform (MAAP) [66], has now provided unprecedented computing resources and satellite images. Deep learning-based methods can extract key structural and spectral features from big data and learn reliable mapping functions across different sensors. Moreover, recent studies have evaluated the influence of registration error on fusion accuracy and concluded that methods learning from patches are more robust to misregistration than traditional rule-based methods that are performed on a per-pixel basis. The abilities to handle big data and tolerate registration errors make deep learning an attractive and powerful tool for linking different sensors at large scales via cloud computing platforms.

VI. Conclusion

Spatiotemporal data fusion is able to provide dense time series images with a high spatial resolution for a wide range of applications. Different from the rule-based spatiotemporal data fusion methods that establish empirical relationships among fine- and coarse-resolution images, we proposed a deep learning network named DenseSTF for blending the Landsat and MODIS images. DenseSTF uses the patch-to-pixel mapping strategy to handle heterogeneous landscapes and accounts for temporal changes via both spatial and temporal dependencies among image pairs. We conducted experiments to assess DenseSTF and compare it with the other deep learning-based methods (i.e., VGG16, STFNET, and EDCSTFN) and existing rule-based methods (i.e., STARFM, ESTARFM, and FSDAF) across three contrasting scenes with different levels of spatial heterogeneity and land cover changes. The deep learning-based model of DenseSTF shows effective and accurate in the fusion of Landsat and MODIS images, particularly in areas with abrupt changes. The modeling strategy and the network structure of the deep learning networks are critical to accurate data fusion. The implementation code for the DenseSTF model is now publicly available via GitHub (https://github.com/sysu-xin-lab/DenseSTF). We welcome researchers and scholars to evaluate and improve the developed DenseSTF model for broad applications.

Acknowledgment

The authors would like to thank the editors and anonymous reviewers for their insightful suggestions that contributed to improving the manuscript.

References

[1] Q. Xin, M. Broich, P. Zhu, and P. Gong, "Modeling grassland spring onset across the western United States using climate variables and MODIS-derived phenology metrics," *Remote Sens. Environ.*, vol. 161, pp. 63–77, May 2015.

[2] L. Tang, S. Zhang, J. Zhang, Y. Liu, and Y. Bai, "Estimating evapotranspiration based on the satellite-retrieved near-infrared reflectance of vegetation (NIRv) over croplands," *GISci. Remote Sens.*, vol. 58, no. 6, pp. 889–913, Aug. 2021.

[3] Y. Liu, S. Zhang, J. Zhang, L. Tang, and Y. Bai, "Assessment and comparison of six machine learning models in estimating evapotranspiration over croplands using remote sensing and meteorological factors," *Remote Sens.*, vol. 13, no. 19, p. 3838, Sep. 2021.

[4] X. Zhu, J. Chen, F. Gao, X. Chen, and J. G. Masek, "An enhanced spatial and temporal adaptive reflectance fusion model for complex heterogeneous regions," *Remote Sens. Environ.*, vol. 114, no. 11, pp. 2610–2623, Nov. 2010.

[5] J. Ju and D. P. Roy, "The availability of cloud-free Landsat ETM+ data over the conterminous United States and globally," *Remote Sens. Environ.*, vol. 112, no. 3, pp. 1196–1211, Mar. 2008.

[6] J. Chen, X. Zhu, J. E. Vogelmann, F. Gao, and S. Jin, "A simple and effective method for filling gaps in Landsat ETM+ SLC-off images," *Remote Sens. Environ.*, vol. 115, no. 4, pp. 1053–1064, Apr. 2011.

[7] Y. Li, C. Huang, J. Hou, J. Gu, G. Zhu, and X. Li, "Mapping daily evapotranspiration based on spatiotemporal fusion of ASTER and MODIS images over irrigated agricultural areas in the Heihe River Basin, Northwest China," *Agricult. Forest Meteorol.*, vols. 244–245, pp. 82–97, Oct. 2017.

[8] G. Camps-Valls, L. Gómez-Chova, J. Muñoz-Marí, J. L. Rojo-Álvarez, and M. Martínez-Ramón, "Kernel-based framework for multitemporal and multisource remote sensing data classification and change detection," *IEEE Trans. Geosci. Remote Sens.*, vol. 46, no. 6, pp. 1822–1835, Jun. 2008.

[9] M. Liu *et al.*, "An improved flexible spatiotemporal data fusion (IFSDAF) method for producing high spatiotemporal resolution normalized difference vegetation index time series," *Remote Sens. Environ.*, vol. 227, pp. 74–89, Jun. 2019.

[10] Z. Malenovský *et al.*, "Scaling dimensions in spectroscopy of soil and vegetation," *Int. J. Appl. Earth Observ. Geoinf.*, vol. 9, no. 2, pp. 137–164, May 2007.

[11] J. Zhou, D. L. Civco, and J. A. Silander, "A wavelet transform method to merge Landsat TM and SPOT panchromatic data," *Int. J. Remote Sens.*, vol. 19, no. 4, pp. 743–757, 1998.

[12] M. Wu and C. Wang, "Spatial and temporal fusion of remote sensing data using wavelet transform," in *Proc. Int. Conf. Remote Sens., Environ. Transp. Eng.*, Jun. 2011, pp. 1581–1584.

[13] B. Huang and H. Zhang, "Spatio-temporal reflectance fusion via unmixing: Accounting for both phenological and land-cover changes," *Int. J. Remote Sens.*, vol. 35, no. 16, pp. 6213–6233, Aug. 2014.

[14] B. Zhukov, D. Oertel, F. Lanzl, and G. Reinhackel, "Unmixing-based multisensor multiresolution image fusion," *IEEE Trans. Geosci. Remote Sens.*, vol. 37, no. 3, pp. 1212–1226, May 1999.

[15] R. Zurita-Milla, J. G. P. W. Clevers, and M. E. Schaepman, "Unmixing-based Landsat TM and MERIS FR data fusion," *IEEE Geosci. Remote Sens. Lett.*, vol. 5, no. 3, pp. 453–457, Jul. 2008.

[16] Z. Niu, "Use of MODIS and Landsat time series data to generate high-resolution temporal synthetic Landsat data using a spatial and temporal reflectance fusion model," *J. Appl. Remote Sens.*, vol. 6, no. 1, Mar. 2012, Art. no. 063507.

[17] W. Zhang *et al.*, "An enhanced spatial and temporal data fusion model for fusing Landsat and MODIS surface reflectance to generate high temporal Landsat-like data," *Remote Sens.*, vol. 5, no. 10, pp. 5346–5368, 2013.

[18] F. Maselli, "Definition of spatially variable spectral endmembers by locally calibrated multivariate regression analyses," *Remote Sens. Environ.*, vol. 75, no. 1, pp. 29–38, Jan. 2001.

[19] L. Busetto, M. Meroni, and R. Colombo, "Combining medium and coarse spatial resolution satellite data to improve the estimation of sub-pixel NDVI time series," *Remote Sens. Environ.*, vol. 112, no. 1, pp. 118–131, Jan. 2008.

[20] R. Zurita-Milla, G. Kaiser, J. G. P. W. Clevers, W. Schneider, and M. E. Schaepman, "Downscaling time series of MERIS full resolution data to monitor vegetation seasonal dynamics," *Remote Sens. Environ.*, vol. 113, no. 9, pp. 1874–1885, Sep. 2009.

[21] C. M. Gevaert and F. J. García-Haro, "A comparison of STARFM and an unmixing-based algorithm for Landsat and MODIS data fusion," *Remote Sens. Environ.*, vol. 156, pp. 34–44, Jan. 2015.

[22] F. Gao, J. Masek, M. Schwaller, and F. Hall, "On the blending of the Landsat and MODIS surface reflectance: Predicting daily Landsat surface reflectance," *IEEE Trans. Geosci. Remote Sens.*, vol. 44, no. 8, pp. 2207–2218, Aug. 2006.

[23] X. Zhu, F. Cai, J. Tian, and T. Williams, "Spatiotemporal fusion of multisource remote sensing data: Literature survey, taxonomy, principles, applications, and future directions," *Remote Sens.*, vol. 10, no. 4, p. 527, Mar. 2018.

[24] T. Hilker *et al.*, "A new data fusion model for high spatial- and temporal-resolution mapping of forest disturbance based on Landsat and MODIS," *Remote Sens. Environ.*, vol. 113, no. 8, pp. 1613–1627, Aug. 2009.

[25] J. Meng, X. Du, and B. Wu, "Generation of high spatial and temporal resolution NDVI and its application in crop biomass estimation," *Int. J. Digit. Earth*, vol. 6, no. 3, pp. 203–218, 2013.

[26] Q. Wang, Y. Tang, X. Tong, and P. M. Atkinson, "Virtual image pair-based spatio-temporal fusion," *Remote Sens. Environ.*, vol. 249, Nov. 2020, Art. no. 112009.

[27] X. Zhu, E. H. Helmer, F. Gao, D. Liu, J. Chen, and M. A. Lefsky, "A flexible spatiotemporal method for fusing satellite images with different resolutions," *Remote Sens. Environ.*, vol. 172, pp. 165–177, Jan. 2016.

[28] X. Li *et al.*, "SFSDAF: An enhanced FSDAF that incorporates sub-pixel class fraction change information for spatio-temporal image fusion," *Remote Sens. Environ.*, vol. 237, Feb. 2020, Art. no. 111537.

[29] D. Guo, W. Shi, M. Hao, and X. Zhu, "FSDAF 2.0: Improving the performance of retrieving land cover changes and preserving spatial details," *Remote Sens. Environ.*, vol. 248, Oct. 2020, Art. no. 111973.

[30] Q. Wang, K. Peng, Y. Tang, X. Tong, and P. M. Atkinson, "Blocks-removed spatial unmixing for downscaling MODIS images," *Remote Sens. Environ.*, vol. 256, Apr. 2021, Art. no. 112325.

[31] Q. Wang and P. M. Atkinson, "Spatio-temporal fusion for daily Sentinel-2 images," *Remote Sens. Environ.*, vol. 204, pp. 31–42, Jan. 2018.

[32] D. Fu, B. Chen, J. Wang, X. Zhu, and T. Hilker, "An improved image fusion approach based on enhanced spatial and temporal the adaptive reflectance fusion model," *Remote Sens.*, vol. 5, no. 12, pp. 6346–6360, 2013.

[33] H. Guan, Y. Su, T. Hu, J. Chen, and Q. Guo, "An object-based strategy for improving the accuracy of spatiotemporal satellite imagery fusion for vegetation-mapping applications," *Remote Sens.*, vol. 11, no. 24, p. 2927, Dec. 2019.

[34] M. Liu, X. Liu, L. Wu, X. Zou, T. Jiang, and B. Zhao, "A modified spatiotemporal fusion algorithm using phenological information for predicting reflectance of paddy Rice in southern China," *Remote Sens.*, vol. 10, no. 5, p. 772, May 2018.

[35] Q. Weng, P. Fu, and F. Gao, "Generating daily land surface temperature at Landsat resolution by fusing Landsat and MODIS data," *Remote Sens. Environ.*, vol. 145, pp. 55–67, Apr. 2014.

[36] J. J. Walker, K. M. de Beurs, and R. H. Wynne, "Dryland vegetation phenology across an elevation gradient in Arizona, USA, investigated with fused MODIS and Landsat data," *Remote Sens. Environ.*, vol. 144, pp. 85–97, Mar. 2014.

[37] D. Xie *et al.*, "An improved STARFM with help of an unmixing-based method to generate high spatial and temporal resolution remote sensing data in complex heterogeneous regions," *Sensors*, vol. 16, no. 2, p. 207, Feb. 2016.

[38] Y. Zhao, B. Huang, and H. Song, "A robust adaptive spatial and temporal image fusion model for complex land surface changes," *Remote Sens. Environ.*, vol. 208, pp. 42–62, Apr. 2018.

[39] B. Huang and H. Song, "Spatiotemporal reflectance fusion via sparse representation," *IEEE Trans. Geosci. Remote Sens.*, vol. 50, no. 10, pp. 3707–3716, Oct. 2012.

[40] X. Liu, C. Deng, S. Wang, G.-B. Huang, B. Zhao, and P. Lauren, "Fast and accurate spatiotemporal fusion based upon extreme learning machine," *IEEE Geosci. Remote Sens. Lett.*, vol. 13, no. 12, pp. 2039–2043, Dec. 2016.

[41] S. P. Boyte, B. K. Wylie, M. B. Rigge, and D. Dahal, "Fusing MODIS with Landsat 8 data to downscale weekly normalized difference vegetation index estimates for central Great Basin rangelands, USA," *GISci. Remote Sens.*, vol. 55, no. 3, pp. 376–399, 2018.

[42] Y. Ke, J. Im, S. Park, and H. Gong, "Downscaling of MODIS one kilometer evapotranspiration using Landsat-8 data and machine learning approaches," *Remote Sens.*, vol. 8, no. 3, p. 215, 2016.

[43] V. Moosavi, A. Talebi, M. H. Mokhtari, S. R. F. Shamsi, and Y. Niazi, "A wavelet-artificial intelligence fusion approach (WAIFA) for blending Landsat and MODIS surface temperature," *Remote Sens. Environ.*, vol. 169, pp. 243–254, Nov. 2015.

[44] H. Song, Q. Liu, G. Wang, L. Hang, and B. Huang, "Spatiotemporal satellite image fusion using deep convolutional neural networks," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 11, no. 3, pp. 821–829, Mar. 2018.
[45] Z. Shao, J. Cai, P. Fu, L. Hu, and T. Liu, "Deep learning-based fusion of Landsat-8 and Sentinel-2 images for a harmonized surface reflectance product," *Remote Sens. Environ.*, vol. 235, Dec. 2019, Art. no. 111425.
[46] Z. Ao, Y. Sun, and Q. Xin, "Constructing 10-m NDVI time series from Landsat 8 and Sentinel 2 images using convolutional neural networks," *IEEE Geosci. Remote Sens. Lett.*, vol. 18, no. 8, pp. 1461–1465, Aug. 2021.
[47] C. Dong, C. C. Loy, K. He, and X. Tang, "Image super-resolution using deep convolutional networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, no. 2, pp. 295–307, Feb. 2016.
[48] Z. Tan, P. Yue, L. Di, and J. Tang, "Deriving high spatiotemporal remote sensing images using deep convolutional network," *Remote Sens.*, vol. 10, no. 7, p. 1066, Jul. 2018.
[49] Z. Tan, L. Di, M. Zhang, L. Guo, and M. Gao, "An enhanced deep convolutional model for spatiotemporal image fusion," *Remote Sens.*, vol. 11, no. 24, p. 2898, Dec. 2019.
[50] X. Liu, C. Deng, J. Chanussot, D. Hong, and B. Zhao, "StfNet: A two-stream convolutional neural network for spatiotemporal image fusion," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 9, pp. 6552–6564, Sep. 2019.
[51] Q. Yuan *et al.*, "Deep learning in environmental remote sensing: Achievements and challenges," *Remote Sens. Environ.*, vol. 241, May 2020, Art. no. 111716.
[52] I. V. Emelyanova, T. R. McVicar, T. G. Van Niel, L. T. Li, and A. I. J. M. van Dijk, "Assessing the accuracy of blending Landsat–MODIS surface reflectances in two landscapes with contrasting spatial and temporal dynamics: A framework for algorithm selection," *Remote Sens. Environ.*, vol. 133, pp. 193–209, Jun. 2013.
[53] Y. Sun, X. Zhang, Q. Xin, and J. Huang, "Developing a multi-filter convolutional neural network for semantic segmentation using high-resolution aerial imagery and LiDAR data," *ISPRS J. Photogramm. Remote Sens.*, vol. 143, pp. 3–14, Sep. 2018.
[54] J. Huang, X. Zhang, Q. Xin, Y. Sun, and P. Zhang, "Automatic building extraction from high-resolution aerial images and LiDAR data using gated residual refinement network," *ISPRS J. Photogramm. Remote Sens.*, vol. 151, pp. 91–105, May 2019.
[55] T. Salimans and D. P. Kingma, "Weight normalization: A simple reparameterization to accelerate training of deep neural networks," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, vol. 29, 2016, pp. 1–9.
[56] J. Yu *et al.*, "Wide activation for efficient and accurate image super-resolution," 2018, *arXiv:1808.08718*.
[57] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014, *arXiv:1409.1556*.
[58] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.
[59] A. A. Jarihani, T. R. Mcvicar, T. G. Van Niel, I. Emelyanova, J. N. Callow, and K. Johansen, "Blending Landsat and MODIS data to generate multispectral indices: A comparison of 'index-then-blend' and 'blend-then-index' approaches," *Remote Sens.*, vol. 6, no. 10, pp. 9213–9238, 2014.
[60] T. Dong *et al.*, "Estimating winter wheat biomass by assimilating leaf area index derived from fusion of Landsat-8 and MODIS data," *Int. J. Appl. Earth Observ. Geoinf.*, vol. 49, pp. 63–74, Jul. 2016.
[61] S. Ioffe, "Batch renormalization: Towards reducing minibatch dependence in batch-normalized models," 2017, *arXiv:1702.03275*.
[62] D. Hendrycks and K. Gimpel, "Gaussian error linear units (GELUs)," 2016, *arXiv:1606.08415*.
[63] T. Dai, J. Cai, Y. Zhang, S.-T. Xia, and L. Zhang, "Second-order attention network for single image super-resolution," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 11057–11066.
[64] L. Zhong, L. Hu, and H. Zhou, "Deep learning based multi-temporal crop classification," *Remote Sens. Environ.*, vol. 221, pp. 430–443, Feb. 2019.
[65] N. Gorelick, M. Hancher, M. Dixon, S. Ilyushchenko, D. Thau, and R. Moore, "Google Earth Engine: Planetary-scale geospatial analysis for everyone," *Remote Sens. Environ.*, vol. 202, pp. 18–27, Dec. 2017.
[66] C. Albinet *et al.*, "A joint ESA-NASA multi-mission algorithm and analysis platform (MAAP) for biomass, NISAR, and GEDI," *Surv. Geophys.*, vol. 40, no. 4, pp. 1017–1027, Jul. 2019.

Zurui Ao received the M.S. and Ph.D. degrees in cartography and geographical information systems from Capital Normal University, Beijing, China, in 2014 and 2018, respectively.

He was a Post-Doctoral Researcher with Sun Yat-sen University, Guangzhou, China, from 2018 to 2021. He is currently an Associate Researcher with the Faculty of Engineering, Beidou Research Institute, South China Normal University, Guangzhou. His research interests include light detection and ranging (LiDAR), deep learning in remote sensing images, and spatiotemporal fusion.

Ying Sun received the B.S. degree in surveying and mapping from Chang'an University, Xi'an, China, in 2005, the M.S. degree in surveying and mapping from Wuhan University, Wuhan, China, in 2007, and the Ph.D. degree in geographical information science from Sun Yat-sen University, Guangzhou, China, in 2014.

She was a Visiting Scholar with the Institute of Space and Earth Information Science, The Chinese University of Hong Kong, Hong Kong, in 2018. She is currently an Associate Professor with the School of Geography and Planning, Sun Yat-sen University. Her research interests include high-resolution remote sensing, deep learning in remote sensing images, and ecology remote sensing.

Xiaoyu Pan received the B.S. degree in geographical information science from South China Normal University, Guangzhou, China, in 2021. She is currently pursuing the M.S. degree with the School of Geography and Planning, Sun Yat-sen University, Guangzhou.

Her research interests include deep learning in remote sensing images and ecology remote sensing.

Qinchuan Xin (Member, IEEE) received the B.S. degree from Peking University, Beijing, China, in 2005, and the Ph.D. degree from Boston University, Boston, MA, USA, in 2012.

From 2012 to 2015, he was a Post-Doctoral Researcher with Tsinghua University, Beijing. He is currently a Professor with the School of Geography and Planning, Sun Yat-sen University, Guangzhou, China. His research interests include ecological remote sensing and terrestrial ecological models.

Dr. Xin also serves as an Associate Editor for *International Journal of Remote Sensing*.