remote sensing

MDPI

Article

Deriving Non-Cloud Contaminated Sentinel-2 Images with RGB and Near-Infrared Bands from Sentinel-1 Images Based on a Conditional Generative Adversarial Network

Quan Xiong [1,2], Liping Di [2], Quanlong Feng [1,3,4], Diyou Liu [1], Wei Liu [1], Xuli Zan [1], Lin Zhang [1], Dehai Zhu [1,3,4], Zhe Liu [1,3,4], Xiaochuang Yao [1,3,4] and Xiaodong Zhang [1,3,4,*]

1 College of Land Science and Technology, China Agricultural University, Beijing 100083, China; xiong@cau.edu.cn (Q.X.); fengql@cau.edu.cn (Q.F.); diyouliu@cau.edu.cn (D.L.); devilweil@cau.edu.cn (W.L.); zanxuli@cau.edu.cn (X.Z.); linzhangcau@cau.edu.cn (L.Z.); zhudehai@cau.edu.cn (D.Z.); liuz@cau.edu.cn (Z.L.); yxc@cau.edu.cn (X.Y.)
2 Center for Spatial Information Science and Systems, George Mason University, 4400 University Dr., Fairfax, VA 22030, USA; ldi@gmu.edu
3 Key Laboratory of Remote Sensing for Agri-Hazards, Ministry of Agriculture, Beijing 100083, China
4 Key Laboratory of Agricultural Land Quality and Monitoring, Ministry of Natural Resources, Beijing 100083, China
* Correspondence: zhangxd@cau.edu.cn; Tel.: +86-139-0113-3526

Abstract: Sentinel-2 images have been widely used in studying land surface phenomena and processes, but they inevitably suffer from cloud contamination. To solve this critical optical data availability issue, it is ideal to fuse Sentinel-1 and Sentinel-2 images to create fused, cloud-free Sentinel-2-like images for facilitating land surface applications. In this paper, we propose a new data fusion model, the Multi-channels Conditional Generative Adversarial Network (MCcGAN), based on the conditional generative adversarial network, which is able to convert images from Domain A to Domain B. With the model, we were able to generate fused, cloud-free Sentinel-2-like images for a target date by using a pair of reference Sentinel-1/Sentinel-2 images and target-date Sentinel-1 images as inputs. In order to demonstrate the superiority of our method, we also compared it with other state-of-the-art methods using the same data. To make the evaluation more objective and reliable, we calculated the root-mean-square-error (RSME), R^2, Kling–Gupta efficiency (KGE), structural similarity index (SSIM), spectral angle mapper (SAM), and peak signal-to-noise ratio (PSNR) of the simulated Sentinel-2 images generated by different methods. The results show that the simulated Sentinel-2 images generated by the MCcGAN have a higher quality and accuracy than those produced via the previous methods.

Keywords: Sentinel-1; Sentinel-2; generative adversarial network; non-cloud contamination; data fusion

Citation: Xiong, Q.; Di, L.; Feng, Q.; Liu, D.; Liu, W.; Zan, X.; Zhang, L.; Zhu, D.; Liu, Z.; Yao, X.; Zhang, X. Deriving Non-Cloud Contaminated Sentinel-2 Images with RGB and Near-Infrared Bands from Sentinel-1 Images Based on a Conditional Generative Adversarial Network. *Remote Sens.* **2021**, *13*, 1512. https://doi.org/10.3390/rs13081512

Academic Editor: Timo Balz

Received: 3 January 2021
Accepted: 12 April 2021
Published: 14 April 2021

1. Introduction

The data of the Sentinel-2 satellite provided by the European Copernicus Earth Observation Program [1] are free and available globally, and they have been widely used in several agricultural applications, such as crop classification [2], cropland monitoring [3], growth evaluation [4,5], and flood mapping [6]. However, as an optical satellite, Sentinel-2 inevitably suffers from cloud and shadow contamination, which can cause a shortage of efficient Earth surface reflectance data for subsequent research [7,8].

Yang et al. [7] and Zhang et al. [9] replaced the contaminated images and created new images with time-close uncontaminated images or the mean of the fore-time phase and the post-time phase. The rationale behind this was that land features should be similar if the time and space of the respective images are close to each other. However, this method places a very high demand on related cloud-free images and is not able to capture changes between the reference data and the target data. In order to solve this issue, some researchers

have tried to fuse other auxiliary optical images with the target optical images to increase the quality of the simulated images [10–13]. Although this method could add some new information to the model, the auxiliary images are still optical data, so if there are continual cloudy days appearing in some places, it is still impossible to collect efficient data.

To solve the cloud contamination problem, we should add some auxiliary data that are able to penetrate clouds. Synthetic Aperture Radar (SAR) could overcome the weakness of optical images. It can work throughout the day and night and under any weather conditions [14]. It has a penetration capacity that captures surface features in spite of clouds [15]. The Sentinel-1 satellite, as an SAR satellite, is provided by the Copernicus Sentinel-1 mission [16], which is free to users, like Sentinel-2. Thus, some researchers started to consider how to use SAR/Sentinel-1 as input data to provide prior information [17]. Ordinarily, this is an SAR-to-optical image translation process [18]. However, SAR and optical remote sensing are fundamentally different in imaging principles, so a captured feature of the same object from these two technologies will be inconsistent. Meanwhile, SAR lacks spectrally resolved measurements, which means it would be a challenge to guarantee the quality of the retrieved spectrum.

With the development of deep learning [19–21], the Generative Adversarial Network (GAN) has received increasing attention in remote sensing due to its superior performance in data generation [22]. Many researchers have since tried to ingest Optical-SAR images into a Conditional Generative Adversarial Network (cGAN) [23], a Cycle-consistent Adversarial Network (CycleGAN) [24], and other GAN models [25,26]. There are two modules in the GAN-based model, one is the Generator, which is used to extract features from input data to generate the simulated images, and the other is the Discriminator, which judges whether the simulated images are real. These two modules, like opponents, compete with each other until the process reaches a trade-off status [27]. We can classify these GAN methods into two categories, supervised and unsupervised. For deriving non-cloud Sentinel-2 images, the cGAN, a supervised GAN, needs a pair of Sentinel-1 and Sentinel-2 images as input. Sentinel-1 is used to provide surface feature information, and Sentinel-2 is able to continually correct the quality of the simulated images [28,29]. The CycleGAN, an unsupervised GAN, only needs a Sentinel-2 dataset and a Sentinel-1 dataset (they could be unpaired), which means this method would have a lighter restraint on data compared with the other methods [24]. Regardless of which kind of methods is used, most researchers have only collected mono-temporal Sentinel-2/Sentinel-1 datasets as input data, which means they tried to learn the relationship between optical images and SAR images at the training stage, but for the inference stage, they only input the Sentinel-1 data into the trained network to generate the simulated Sentinel-2 data [30–33]. This would demand that Sentinel-1 has the same distinguishing ability as Sentinel-2 for different surface objects. Otherwise, the simulated images could not guarantee that enough details are simulated. Unfortunately, some surface objects always have similar backscattering properties, which makes it difficult to distinguish them using SAR data [31]. Therefore, this kind of input data is a little too simple to guarantee a satisfactory accuracy.

In order to keep more real Earth surface details in the simulated optical images, Li et al. [34] and Wang et al. [35] started to pay attention to the corrupted optical images. These researchers tried to add the corrupted optical images as a kind of additional input data along with the Optical-SAR image pairs mentioned above to the network [36,37]. Adding the corrupted optical data, compared with other input data, is an efficient way to improve the accuracy of the simulated images. Ordinarily, the simulated images could save some textural and color details from the uncontaminated part of the corrupted images. However, how enough information can be learned if the corrupted optical images are covered by a large amount of clouds is still an open question. As we know, if we want to use deep learning, we often need to split the whole image into many small patches, such as images in 256 $\times$ 256 [38] or 128 $\times$ 128 [39]. That means that, even if we have filtered the cloud percentage and have selected some images with scattered clouds, such as these corrupted images, there is still a possibility that some split small images are full of thin

or thick clouds. Meanwhile, this approach needs these corrupted images as input, which means it is not able to generate cloud-free optical images at other time phases when there are no optical images captured by satellite [25,40].

In this study, we increased the channels of the cGAN [23] to make the model able to exploit multi-temporal Sentinel-2/Sentinel-1 image pairs, which is the MCcGAN. The structure of this paper is as follows: In Section 2, we introduce the structure and loss functions of the cGAN we used and explain how the three different datasets can be built. In Section 3, we elaborate how we implemented the experiments to assess different methods. Finally, Sections 4 and 5 discuss the experimental results and describe future work.

2. Materials and Methods

2.1. Data Preprocessing

In order to keep the simulated optical images with more realistic details, we built a multi-temporal Sentinel-2/Sentinel-1 image dataset model. The goal of this model was to learn the optical-SAR spectral relationship between Sentinel-1 and Sentinel-2 at Time I and textural features from Sentinel-2 at Time I, and then to use Sentinel-1 at Time II to simulate Sentinel-2 at Time II.

Sentinel-1 and Sentinel-2 were downloaded from https://scihub.copernicus.eu/dhus/ (accessed on 10 August 2020). To avoid additional errors caused by resampling, we selected Sentinel-1 and Sentinel-2 data products with the same spatial resolution. For Sentinel-1, we chose the Ground Range Detected (GRD) products in Interferometric Wide swath mode (IW), and we only used the two bands, VV (vertical transmit and vertical receive) and VH (vertical transmit and horizontal receive), with a pixel space of 10 m. For Sentinel-2, we chose the Level-1C product (Top of Atmosphere (TOA) reflectance), and we only used red, green, and blue bands with a resolution of 10 m. The whole flowchart of data preprocessing is shown in Figure 1.

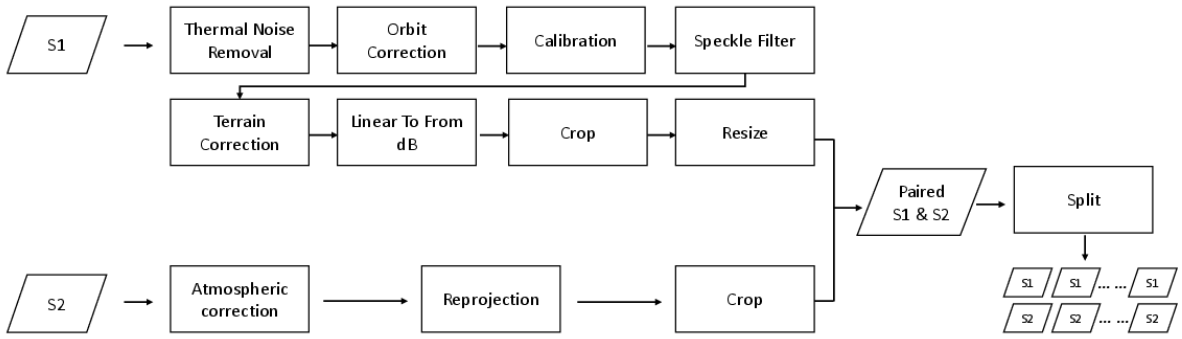

Figure 1. The flowchart of the data preprocessing.

The preprocessing of the Sentinel-1 images includes Thermal Noise Removal, Orbit Correction, Calibration, Speckle Filter, Terrain Correction, Backscattering Coefficient Conversion to a dB Value, Crop, and Resize. The preprocess of Sentinel-2 mainly includes Atmospheric Correction, Reprojection, and Crop. Reprojection projects Sentinel-2 to the World Geodetic System 1984 (WGS84), which is the same as Sentinel-1's coordinate system. Crop and Resize gives these two data the same geographical space and the same number of pixels. After that, we acquired the paired images, and the Split operation was conducted to split large images into many small patch images (size: 256 × 256; stride: 128), which means each small patch image had half of an overlapped area with an adjacent small patch image. A multi-temporal dataset was then built. All professional preprocesses in Figure 1 were conducted with the free software named SNAP (http://step.esa.int/main/ (accessed on 12 August 2020)). The Crop, Resize, Split operations were implemented based on the Python code provided by [31] (https://github.com/whu-csl/WHU-SEN-City (accessed on 14 August 2020)).

2.2. Dataset A

Dataset A was used to research the results of models when it came to the accuracy of the spatial transfer. The data were taken from Gansu Province, China. The coordinates were 45.1467° N, 85.7274° E, 44.1653° N, 87.1220° E. The times of the Sentinel-1/Sentinel-2 image pairs are shown in Table 1. In order to pair S1/S2, we made an appropriate crop operation, so they were slightly smaller than the original images downloaded from the official website.

Table 1. The times of the Sentinel-1/Sentinel-2 image pairs.

	S1 I	S2 I	S1 II	S2 II
Date	29 August 2018	1 September 2018	10 September 2018	11 September 2018

Figure 2 shows these four images (size: 13,238 × 8825) and how we selected the training dataset. We split the whole image into two parts (the red line). We chose the part above the red line as the training patch set. This part occupies about 90% of the whole image. We then split this part into many small tiles, as mentioned in Section 2.1, to build the multi-temporal dataset or the mono-temporal dataset. For methods that need multi-temporal information, we ingested the multi-temporal dataset into the model, and there were 6018 training pairs. For methods that are only suitable for mono-temporal information, we split the multi-temporal dataset into two parts (S1 I/S2 I and S1 II/S2 II) and then put these two parts into the model, and there were 12,036 training pairs. In a word, the principle was to input the same information into these models to obtain the results, thus maintaining fairness in comparing different approaches. The formats of Sentinel-1 and Sentinel-2 were JPG. The pixel values of Sentinel-2 were the reflectance, and the pixel values of Sentinel-1 were the backscattering coefficients.

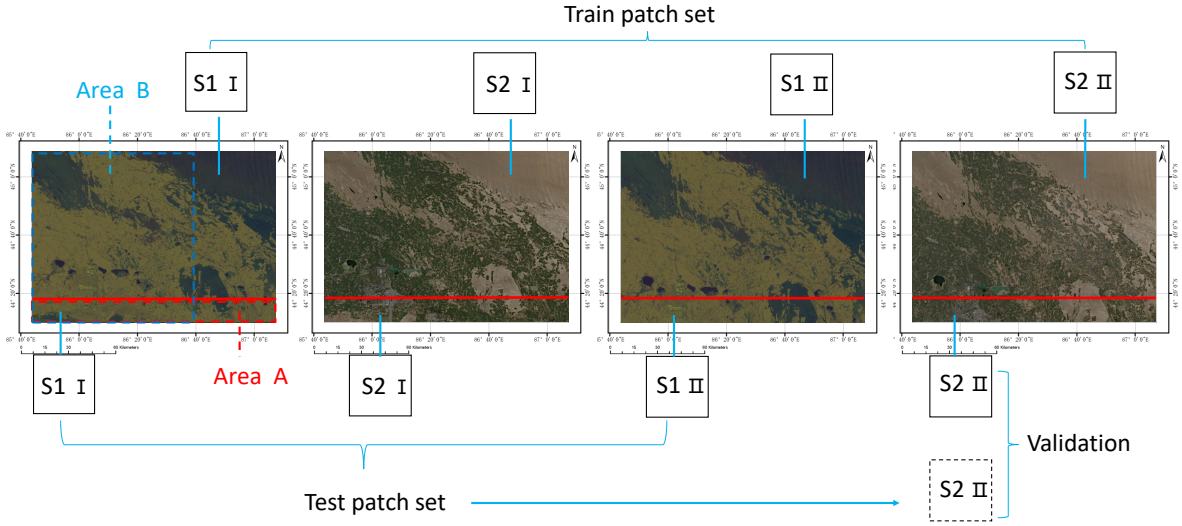

Figure 2. The train patch set and test patch set (the solid line box: real data, the dotted line box: simulated data, Area A: validation area for spatial transfer, Area B: validation area for time transfer).

2.3. Dataset B

Dataset B was used to research the results of models in terms of the accuracy of the time transfer. We downloaded and produced two other paired Sentinel-1/Sentinel-2 images at Time III and Time IV. The spatial range (size: 8001 × 8704) was the subset (because there were some clouds in the east of S2 III, we only used the cloud-free area, which we called Area B) of the geographical space of the images shown in Figure 2. The training dataset was the same as Dataset A, and we again used S1 I/S2 I as reference data to generate the

simulated S2 at Time III and Time IV. The times of these two test image pairs are shown in Table 2. The meaning of the pixel values and the formats of Sentinel-1 and Sentinel-2 in this dataset were the same as those in Dataset A.

Table 2. The times of the Sentinel-1/Sentinel-2 image pairs.

	S1 III	S2 III	S1 IV	S2 IV
Date	22 September 2018	21 September 2018	4 October 2018	6 October 2018

2.4. Dataset C

The above datasets were in regard to a single geographical space and could not show the advantages and disadvantages of each method applied on a global scale. At the same time, in addition to the red, green, and blue bands, the near-infrared band is also of irreplaceable importance for agricultural applications. Therefore, Dataset C was re-selected through the Google Earth Engine (GEE) to acquire a training set and a validation set. Their spatial distributions are shown in Figure 3, and dates are shown in Table A7. The total number of S1/S2 image pairs was 38. The space of each region of interest was less than or equal to $0.5° \times 0.5°$. We used 'Sentinel-1 SAR GRD: C-band Synthetic Aperture Radar' and 'Sentinel-2 MSI: MultiSpectral Instrument, Level-2A' products provided by GEE to acquire Sentinel 1 data and Sentinel 2 data, respectively. Since these two data products have been preprocessed, compared with Figure 1, we only needed to download multi-temporal Sentinel 1 (VV and VH bands) and Sentinel 2 (red, green, blue, and near-infrared bands) data for each region of interest, and we then carried out subsequent splitting to obtain the dataset required by these deep learning models. For the multi-temporal models, there were 6355 training pairs. For the mono-temporal models, to ensure the consistency of the input data, we split the multi-temporal data into mono-temporal data, and there were 12,710 training pairs. As for the validation dataset, due to a lack of memory, remote sensing images were also split in advance before they were input into the model for inference, and the results were finally mosaicked together for quantitative evaluation. The spatial distribution of validation data is also shown in Figure 3. Number labels (1–5) were used to distinguish the different areas. The formats of Sentinel-1 and Sentinel-2 were TIFF. The pixel values of Sentinel-2 were the reflectance, and the pixel values of Sentinel-1 were the backscattering coefficients after the decibel.

Figure 3. The distribution of training data and validation data (red represents the training set, and green represents the validation data).

2.5. Conditional Generative Adversarial Network

The cGAN [41] is an extension of the original GAN [22], which is made of two adversarial neural network modules, the Generator (G) and the Discriminator (D). The G attempts to extract features from real images in Domain X to generate simulated images in Domain Y to fool the D, whereas the D is trained from real images in Domains X and Y to discriminate the input data as synthetic or real. The flowchart of the cGAN is shown in Figure 4.

The loss of the G comes from two parts, GANLoss and L1Loss. GANLoss expresses whether the simulated images y′ are able to fool the D (cause the D to judge the simulated images as real images). L1Loss presents the distance between real images y and simulated images y′. The G tries to make these two loss functions' values close to zero, which means the simulated results are more realistic. Contrarily, the D is in charge of discriminating the real images y as true and the simulated images y′ as false, which is DLoss. Similar to the G, the D also tries to make the DLoss function's value close to zero to improve its own distinguishing capacity. The objective function for the cGAN ($\mathcal{L}_{cGAN}$(G,D)) is expressed in Equation (1):

$$\mathcal{L}_{cGAN}(G,D) = \mathbb{E}_{x,y\sim p_{data}(x,y)}[\log D(x,y)] + \mathbb{E}_{x\sim p(x),z\sim p(z)}[\log(1-D(x,G(x,z)))] \quad (1)$$

where $\mathbb{E}$ and log are expectation and logarithmic operators, respectively, p is the distribution of the images, and z is a random noise vector that follows a prior known distribution $p(z)$, typically uniform or Gaussian.

Ordinarily, as the L1Loss shown in Figure 4, an L1 norm distance loss is added to the objective function of the cGAN to make the simulated images more similar to the real images, as shown in Equation (2):

$$G^* = \arg\min_G \max_D \mathcal{L}_{cGAN}(G,D) + \lambda \mathcal{L}_{L1}(G) \quad (2)$$

where λ is a regularization weight, and $\mathcal{L}_{L1}$ is defined as Equation (3).

$$\mathcal{L}_{L1}(G) = \mathbb{E}_{x,y\sim p_{data}(x,y),z\sim p(z)}[\|y - G(x,z)\|_1] \quad (3)$$

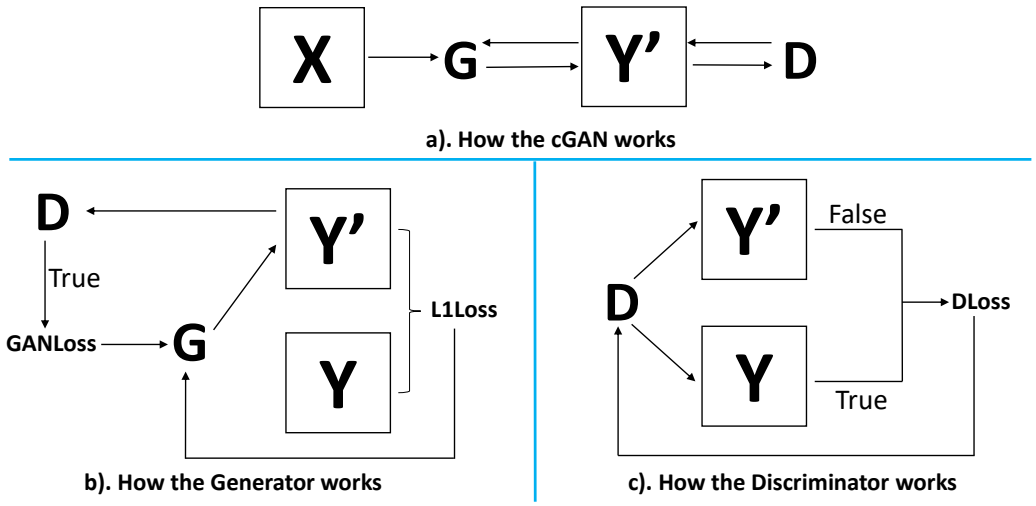

Figure 4. The flowchart of the cGAN (D: Discriminator, G: Generator).

2.6. The Structure of the Proposed Method

We designed the MCcGAN based on [31]. The structure is shown in Figure 5. We used U-net [42] with 16 layers to build the Generator. The input data were the concatenation of Sentinel-1's VV and VH at Times I and II, respectively, and the Sentinel-2's RGB bands at Time I. The output data were the simulated Sentinel-2 images with RGB bands at Time II. The peculiarity of U-net is concatenating the ith Encoder's output and the $(n-i)$th Decoder's output as new intermediate inputs to be ingested into the next Decoder. The network is able to learn features in a low dimension and a high dimension simultaneously, which can lead to results with more details.

The Discriminator was built with the PatchGAN [23], which includes 5 Encoders. To make the D distinguish between the real images and the simulated images generated by the G, we marked the concatenation of the input data of the G and the real images as true and, similarly, marked the concatenation of the input data of the G and the simulated images as false. We then put these two kinds of data into the D to train the model.

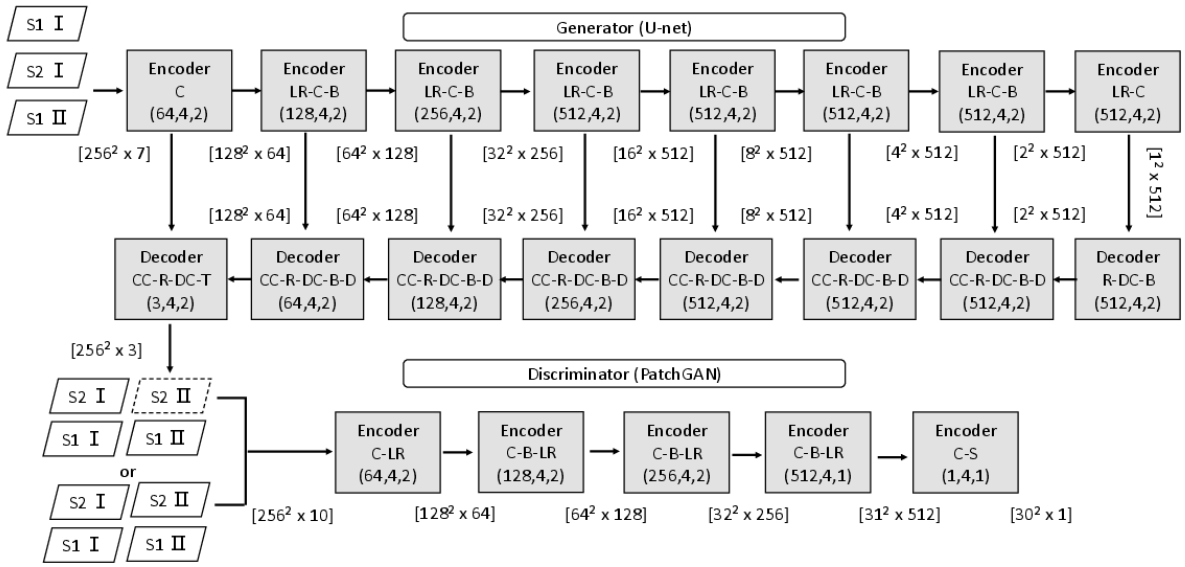

Figure 5. Structure of the Generator and Discriminator (C: Convolution, LR: Leaky ReLU, B: Batch Normalization, R: ReLU, DC: Deconvolution, CC: Concatenation, D: Dropout, T: Tanh, S: Sigmoid. The three numbers in the parentheses denote the number of filters, filter size, and stride, respectively. The numbers in the brackets indicate the size and the number of features of the images).

2.7. Other Methods for Comparison

Based on a review of related literature, we selected the following methods to compare with the MCcGAN. The MTcGAN [43] is also able to ingest a multi-temporal dataset, but its main deep learning network is a Residual Network (ResNet). The deep convolutional spatiotemporal fusion network (DCSTFN) [10] has been used for Optical-Optical image pairs. We implemented the generative adversarial network based on the structure of the DCSTFN cGAN model. We refer to this as DCSTFN-OS for Optical-SAR translation tasks. We also tried to add the ResNet to DCSTFN-OS because the ResNet is able to extract more details [44]. We refer to this as RDCSTFN-OS. These three methods are all suitable for multi-temporal datasets. We also chose some state-of-the-art methods suitable for mono-temporal datasets to make comparisons, e.g., the CycleGAN [24], which is able to learn information from training sets without pairs. The S-CycleGAN [31] is a modified version of the CycleGAN that can be trained with pairs of training sets. pix2pix [40] is a typical conditional generative adversarial network.

2.8. Evaluation Metrics

The root-mean-square-error (RMSE) can measure the difference between the real and simulated values. The formula is defined as follows:

$$RMSE = \sqrt{\frac{\sum_{i=1}^{N}(y_i - \hat{y}_i)^2}{N}} \tag{4}$$

where y_i and $\hat{y}_i$ are the real and simulated values for the ith pixel, respectively, and N is the number of pixels. The smaller RMSE is, the more similar the two images are.

The coefficient of determination (R^2) can measure how close the data are fitted to the regression function. It can be defined as Equation (5):

$$R^2 = 1 - \frac{\sum_{i=1}^{N}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{N}(y_i - \overline{y}_i)^2} \tag{5}$$

where y_i and $\hat{y}_i$ are the real and simulated values for the ith pixel, respectively, $\overline{y}$ represents the mean of the real values, and N is the number of pixels. The closer it is to 1, the better the simulation is.

The Kling–Gupta efficiency (KGE) [45] can also evaluate the efficiency between the real data and simulated data. The formula is defined as follows:

$$KGE = 1 - \sqrt{(r-1)^2 + (\frac{\sigma_{\hat{y}}}{\sigma_y} - 1)^2 + (\frac{\mu_{\hat{y}}}{\mu_y} - 1)^2} \tag{6}$$

where r represents the correlation coefficient between the real data and the simulated data, $\sigma_{\hat{y}}$ and σ_y denote the standard deviation of the simulated values and real values, respectively, and $\mu_{\hat{y}}$ and μ_y denote the mean of the simulated and real values, respectively. The simulation is better if KGE is closer to 1.

The structural similarity index (SSIM) [46] can measure the structural similarity between the real image and the simulated image. It can be defined as Equation (7):

$$SSIM = \frac{(2\mu_y\mu_{\hat{y}} + C_1)(2\sigma_{y\hat{y}} + C_2)}{(\mu_y^2 + \mu_{\hat{y}}^2 + C_1)(\sigma_y^2 + \sigma_{\hat{y}}^2 + C_2)} \tag{7}$$

where $\sigma_{y\hat{y}}$ represents the covariance between the real and simulated values, C_1 and C_2 are the constants to enhance the stability of SSIM, and the other parameters are the same as those in the formulas mentioned above. The value range of SSIM is from -1 to 1. The closer it is to 1, the better the simulated image is.

The peak signal-to-noise ratio (PSNR) [47] is a traditional image quality assessment (IQA) index. A higher PSNR generally demonstrates that the image is of higher quality. The formula can be defined as follows:

$$PSNR = 10\lg(\frac{255^2}{MSE}) \tag{8}$$

$$MSE = \frac{\sum_{i=1}^{N}(y_i - \hat{y}_i)^2}{N} \tag{9}$$

where MSE is the acronym of the mean-square-error, and the other parameters are the same as those in the formulas mentioned above.

The spectral angle mapper (SAM) [48] is used to calculate the similarity between two arrays, and the result can be regarded as the cosine angle between two arrays. The smaller the output value, the more the two arrays match, and the more similar they are.

3. Experiment and Results

This section shows the results of different methods regarding the spatial transfer, the time transfer, and the global scale.

3.1. Experiment on Spatial Transfer

In this experiment, we validated whether the proposed method and other methods are able to simulate Sentinel-2 images with high quality in the target area. The training dataset has been introduced in Section 2.2, and the test dataset used in this experiment is shown in Figure 2. We chose the area (Area A, about 10 % of the whole image) below the red line to produce the test dataset. The test dataset containing S1 I, S2 I, and S1 II was ingested into the multi-temporal models to generate the simulated S2 II. For the models that were trained with a mono-temporal dataset, we only ingested S1 II to generate the simulated image. Finally, we estimated the similarity between the real S2 II and the simulated S2 II. The evaluation metrics of different methods for different bands in Area A are shown in Tables 3–5. The SAM of different methods in Area A is shown in Table 6.

Table 3. The evaluation metrics of different methods for the red band in Area A at Time II (the bold result is the best among the different metrics, MCcGAN: Multi-channels Conditional Generative Adversarial Network, MTcGAN: Multi-temporal Conditional Generative Adversarial Network, DCSTFN-OS: Deep Convolutional Spatiotemporal Fusion Network for SAR-to-Optical Task, RDCSTFN-OS: Residual Deep Convolutional Spatiotemporal Fusion Network for SAR-to-Optical Task, CycleGAN: Cycle-consistent Adversairal Network, S-CycleGAN: Supervised Cycle-consistent Adversairal Network, RMSE: root-mean-square-error, KGE: Kling Gupta efficiency, SSIM: structural similarity index, PSNR: peak signal-to-noise ratio).

	MCcGAN	MTcGAN	DCSTFN-OS	RDCSTFN-OS	CycleGAN	S-CycleGAN	pix2pix
RMSE	**5.5057**	5.6978	5.7848	5.5204	10.2395	7.8483	7.6142
R^2	**0.8830**	0.8747	0.8708	0.8824	0.5954	0.7623	0.7762
KGE	**0.8631**	0.8310	0.8461	0.8511	−0.5826	0.6575	0.6955
SSIM	**0.8916**	0.8885	0.8826	0.8848	0.3078	0.6394	0.6833
PSNR	**30.7792**	30.0434	30.1811	30.3719	18.5376	26.1466	26.5571

Table 4. The evaluation metrics of different methods for the green band in Area A at Time II (the bold result is the best among the different metrics).

	MCcGAN	MTcGAN	DCSTFN-OS	RDCSTFN-OS	CycleGAN	S-CycleGAN	pix2pix
RMSE	**4.8704**	5.3527	5.4037	4.9179	9.8007	7.3618	7.1411
R^2	**0.8322**	0.7973	0.7934	0.8289	0.3207	0.6167	0.6393
KGE	**0.8529**	0.7880	0.8173	0.8270	−0.4411	0.5776	0.6132
SSIM	**0.9085**	0.9030	0.8972	0.9048	0.4205	0.6726	0.7173
PSNR	**32.9836**	31.9480	32.0819	32.6580	21.4718	27.7347	28.2506

Table 5. The evaluation metrics of different methods for the blue band in Area A at Time II (the bold result is the best among the different metrics).

	MCcGAN	MTcGAN	DCSTFN-OS	RDCSTFN-OS	CycleGAN	S-CycleGAN	pix2pix
RMSE	4.6866	5.2007	5.0652	**4.6793**	9.6357	7.2521	6.8632
R^2	0.8196	0.7779	0.7893	**0.8201**	0.2375	0.5681	0.6132
KGE	0.8288	0.7749	0.8088	**0.8294**	−0.3844	0.4963	0.5651
SSIM	**0.8874**	0.8841	0.8781	0.8871	0.3771	0.6382	0.6890
PSNR	**33.7399**	32.7198	32.9846	33.5258	22.7431	27.9967	28.7693

Table 6. The spectral angle mapper (SAM) of different methods in Area A at Time II (the bold result is the best).

	MCcGAN	MTcGAN	DCSTFN-OS	RDCSTFN-OS	CycleGAN	S-CycleGAN	pix2pix
SAM	**0.2069**	0.2269	0.2244	0.2278	0.7496	0.4206	0.4002

Tables 3–6 indicate that the proposed method has the best simulated accuracy under most circumstances. Meanwhile, the results also demonstrate that the proposed method can be used for spatial transfer learning. The results of the multi-temporal models (MCcGAN, MTcGAN, DCSTFN-OS, and RDCSTFN-OS) and the mono-temporal modesl (CycleGAN, S-CycleGAN, and pix2pix) also show that the models with a multi-temporal dataset are better than those with a mono-temporal dataset. Details for judging the simulated results are shown in Figure 6.

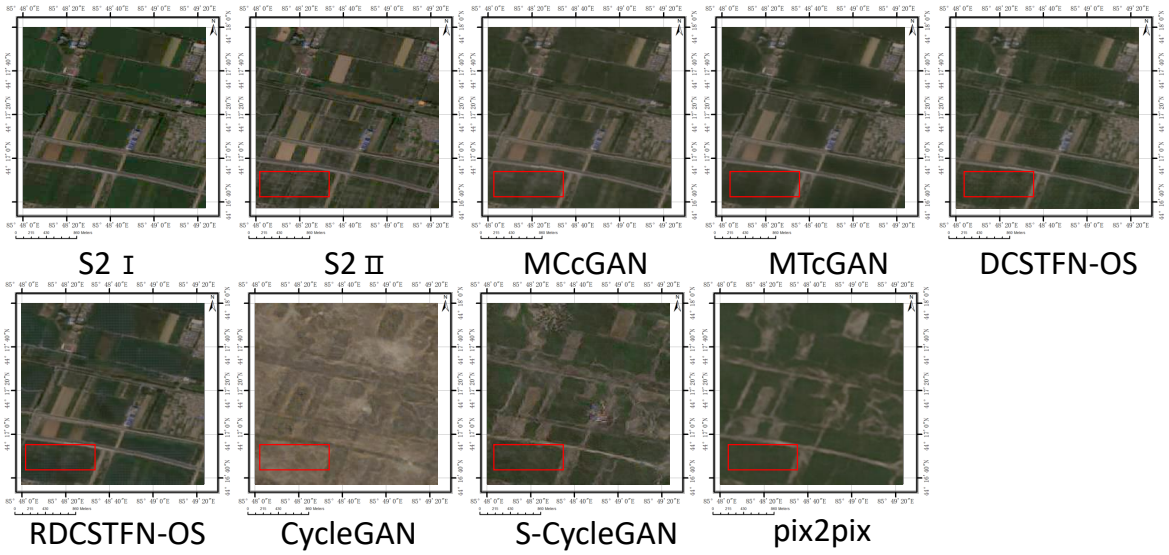

Figure 6. Comparison of different methods (image size: 256 × 256, MCcGAN: Multi-channels Conditional Generative Adversarial Network, MTcGAN: Multi-temporal Conditional Generative Adversarial Network, DCSTFN-OS: Deep Convolutional Spatiotemporal Fusion Network for SAR-to-Optical Task, RDCSTFN-OS: Residual Deep Convolutional Spatiotemporal Fusion Network for SAR-to-Optical Task, CycleGAN: Cycle-consistent Adversairal Network, S-CycleGAN: Supervised Cycle-consistent Adversairal Network).

The surface object in the red rectangle exhibits an apparent change between the reference image S2 I and target real image S2 II. As we see, the MCcGAN was able to simulate the image with more detail compared to the other methods. The mono-temporal models could simulate some high dimensional features, but they missed some details (blurring the results), which is consistent with the results in Tables 3–6. The CycleGAN mainly tries to transfer an image's style with an unpaired dataset; thus, for the Optical-SAR transfer task, it enables SAR style images to look like optical images, but it is hard to guarantee pixel-wise accuracy.

3.2. Experiment of Time Transfer

In this experiment, we intended to explore the change in accuracy with different time intervals for the MCcGAN. We also ran all the methods in this experiment to see which one had the highest accuracy. The experimental data are Dataset B.

The RMSE and SSIM results of different methods are shown in Figures 7 and 8. The evaluation metrics of the different methods for different bands in Area B at Time III are shown in Tables A1–A3. The evaluation metrics of different methods for different bands in Area B at Time IV are shown in Tables A4–A6. Regardless of the band, the two figures demonstrate that the proposed method's simulated image is the best. In terms of the quality of simulated spectra, our method is also the best, as Table 7 shows. However, when we compared the results of the MCcGAN at Time III and Time IV, overall, we found that the results at Time III are better than those at Time IV, which means the proposed method might be sensitive to the time interval between the reference Optical-SAR image pairs and the target image.

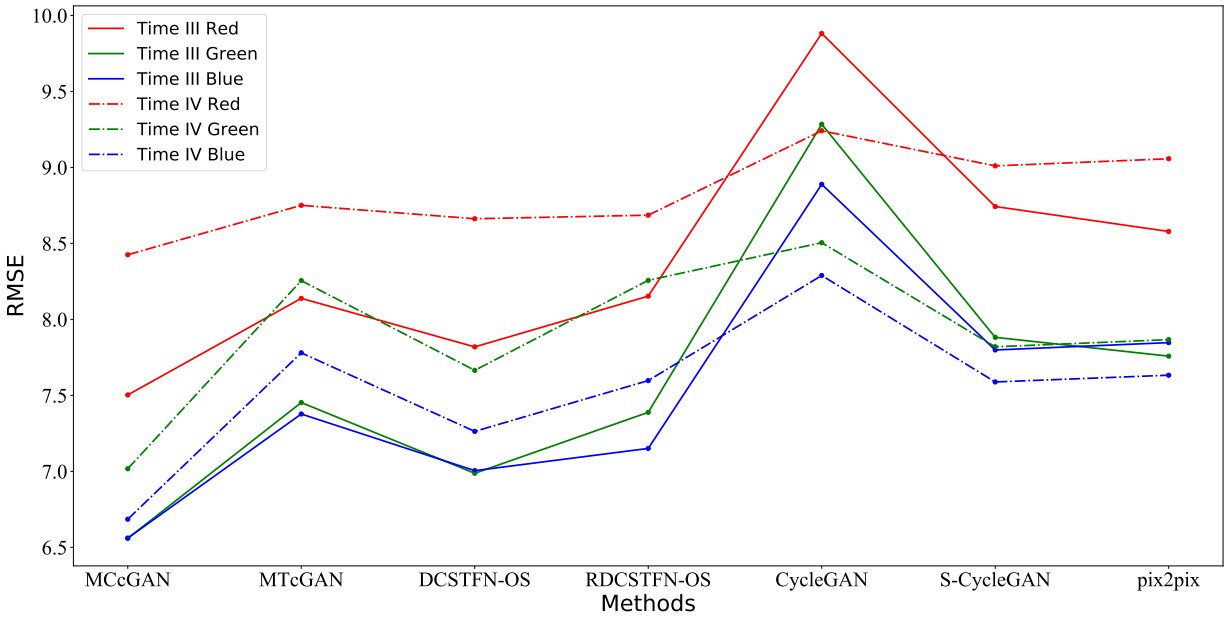

Figure 7. The RMSE results of different methods for different bands in Area B at Times III and IV.

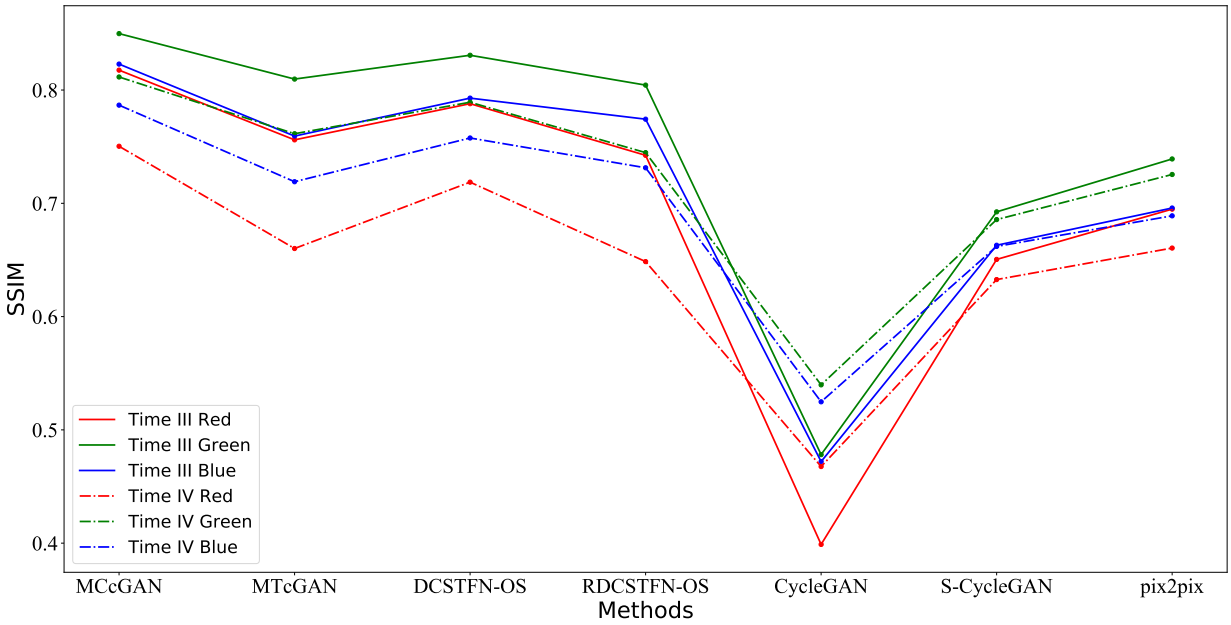

Figure 8. The SSIM results of different methods for different bands in Area B at Times III and IV.

Table 7. The spectral angle mapper (SAM) of different methods in Area B at Times III and IV (the bold result is the best).

	MCcGAN	MTcGAN	DCSTFN-OS	RDCSTFN-OS	CycleGAN	S-CycleGAN	pix2pix
Time III	**0.2559**	0.3424	0.2981	0.3407	0.6586	0.3515	0.3338
Time IV	**0.3212**	0.4329	0.3615	0.4333	0.5628	0.3718	0.3754

3.3. Experiment of Global Scale

In order to quantitatively evaluate the results of different methods on the global scale, we used Dataset C to re-train the MCcGAN and calculated the mean evaluation metrics of each band (red, green, blue, and near-infrared) of the validation data, as Table 8 shows. The methods whose results rank first and second among many methods are shown. It

is not difficult to see that the multi-temporal modeling method is better than the mono-temporal modeling method, and the method proposed in this paper is more stable in most cases. What is more, the RMSE in Table 8 is far larger than in the previous tables, which is because the data format of the results in this experiment was TIFF, and the reflectance value was from 0 to 10,000. However, the experimental results in the previous section were stored in JPG, and the reflectance value ranged from 0 to 255. In order to verify the availability of these simulated images for subsequent applications, we calculated the Normalized Difference Vegetation Index (NDVI) and its evaluation metrics, as Table A8 shows. The results show that our method is always the best or second-best choice, which also demonstrates that our method has satisfactory stability.

Table 8. The mean evaluation metrics of different methods for four bands in different areas (the bold result is the best among the different metrics, and the underlined result is the second best).

		MCcGAN	MTcGAN	CycleGAN	S-CycleGAN	pix2pix
Area 1	RMSE	**133.1711**	142.1898	157.2520	159.9680	147.1722
	R^2	**0.8591**	0.8316	0.8013	0.7862	0.8316
	KGE	**0.7684**	0.7206	−0.1016	0.1764	0.1604
	SSIM	**0.9766**	0.9693	0.9169	0.9339	0.9487
	PSNR	**48.6183**	47.7088	42.5771	44.0798	44.7184
	SAM	**0.1874**	0.2003	0.3426	0.3746	0.2893
Area 2	RMSE	**130.4381**	131.0418	162.7805	164.9328	155.1990
	R^2	**0.9234**	0.9220	0.8716	0.8641	0.8836
	KGE	0.7680	**0.7714**	−0.1890	0.0396	0.1780
	SSIM	0.9769	**0.9782**	0.8912	0.9227	0.9424
	PSNR	47.6960	**47.7433**	40.2789	42.4109	43.2338
	SAM	0.1994	**0.1813**	0.3102	0.3278	0.2665
Area 3	RMSE	**111.6861**	115.2743	145.3813	154.1060	146.4545
	R^2	**0.8064**	0.8058	0.6857	0.6306	0.6930
	KGE	**0.8205**	0.7459	−0.0302	0.0508	0.1351
	SSIM	**0.9917**	0.9909	0.9634	0.9604	0.9607
	PSNR	**53.1270**	52.2580	46.3271	45.6118	45.5786
	SAM	**0.1045**	0.1277	0.2677	0.2337	0.2751
Area 4	RMSE	**113.4953**	130.2571	163.7737	163.4896	159.5248
	R^2	**0.9112**	0.8949	0.8093	0.8093	0.8160
	KGE	**0.8337**	0.8104	0.0078	0.0799	0.1383
	SSIM	**0.9929**	0.9899	0.9094	0.9276	0.9357
	PSNR	**52.0253**	50.7773	41.0806	42.6810	42.9448
	SAM	**0.0892**	0.0942	0.3167	0.3076	0.3053
Area 5	RMSE	**136.0845**	137.7683	157.2821	169.0275	150.9014
	R^2	**0.8859**	0.8823	0.8325	0.7797	0.8528
	KGE	0.4441	**0.4562**	−0.3786	0.0101	0.1786
	SSIM	**0.9453**	0.9452	0.9073	0.9064	0.9409
	PSNR	43.9744	**44.1947**	42.0800	42.0878	43.9135
	SAM	0.3560	**0.3367**	0.3700	0.4037	0.3678

4. Discussion

The multi-temporal models were not only able to convert the style of the images from SAR to Optical, but were also able to save more details. They could simulate optical images with a higher quality compared to the mono-temporal models, especially the proposed method (MCcGAN), whose advantages are shown in Section 3. Using the mono-temporal models, due to the difference between SAR data and optical data in ground object reflection, it is more difficult to generate optical images with high quality only from mono-temporal SAR. However, generally, these multi-temporal models are sensitive to the time interval

between the reference data and the simulated data, which means the MCcGAN cannot guarantee simulation quality if the acquisition date of the reference cloud-free optical image is far away from the target date. We could not quantitatively estimate the relationship between the time interval and the simulation quality in this study, and this is also a common issue in other related literature. In future work, we will use datasets with more phases to explore the relationship between the time interval and the simulation quality.

Compared with the proposed method, the mono-temporal models (CycleGAN [24], S-CycleGAN [31], and pix2pix [40]) were also able to convert the style of images from SAR to Optical, but they could not recover more details. Their advantage is that they do not need reference paired images, which means they can play a significant role if it is hard to acquire reference optical images with high quality. Moreover, we think that the CycleGAN is not suitable for this kind of task because, while its advantage is an unsupervised transfer, it is not able to retain pixel-wise accuracy.

Although both the MCcGAN and the MTcGAN [43] are multi-temporal models, they have their own advantages. Table 8 shows that the MCcGAN is better than the MTcGAN in most validation areas, except for Area 2. We checked the main types of the surface object in these five areas. Area 2 contains some towns and mountains, and other areas are cropland. Therefore, when comparing these two models, the MTcGAN is more suitable for generating cloud-free Sentinel-2-like images in the towns and mountains, and the MCcGAN is better for generating cloud-free Sentinel-2-like images in the cropland.

The loss functions used in this paper are GANLoss, L1Loss, and DLoss, which are loss functions commonly used in typical conditional generative adversarial networks. In the reference [36], Gao et al. proposed that a perceptual loss function is able to generate results with better visual perception, because this function is designed to measure high dimensional features such as color and shape. However, we mainly updated the network with low dimensional features (the difference in pixel values between the simulated images and the real images). Next, we would add this function to MCcGAN to see whether it is useful for our model.

In fact, we only used two types of temporal information. In the future, we hope to add more time-series data as a reference to simulate the target data. We want to explore whether the cGAN is able to capture changes according to the reference data and transfer it to the simulated stage to retain results with high quality.

In order to obtain simulated optical images with more detail, we also intend to use corrupted real optical images in our model in the future. The results are expected to retain details from cloud-free areas in corrupted real optical images. For corrupted areas, we hope that the model can recover the details from the multi-temporal reference data. In this way, we think the simulation will have a higher quality. To evaluate the results, it will be necessary to add absolute validation with ground truth data. Therefore, collecting ground truth data from the Radiometric Calibration Network portal (RadCalNet, https://www.radcalnet.org/#!/ (accessed on 10 December 2020)) will be our next task.

5. Conclusions

In this paper, we added new channels to the cGAN to obtain the MCcGAN, which is able to learn information from multi-temporal paired images of Sentinel-1 and Sentinel-2. We downloaded and processed original Sentinel series images from the official website and then produced them as a multi-temporal dataset used in this paper. Meanwhile, the global dataset was built based on the GEE. To explore the advantage of the proposed method, we designed three experiments to make comparisons with other state-of-the-art methods. In order to quantitatively assess the results, we used popular statistical metrics, including RMSE, R^2, KGE, SSIM, PSNR, and SAM. The results in the first experiment illustrated that the proposed method can be trained in one place and then used in another place. The results in the second experiment showed that the proposed method is sensitive to the time interval between the reference data and the simulated data. It would be better to keep the time interval as narrow as possible when using the MCcGAN. The last experiment

proved that the proposed method is applicable on a large scale. The proposed method not only succeeded in transferring the style of the images from SAR to Optical but also recovered more details. It is superior to other methods. We think that our method can play an important role in Optical-SAR transfer tasks, data filling in crop classification, and other research.

Author Contributions: Conceptualization, Q.X.; Methodology, Q.X.; Software, W.L., D.L. and Q.X.; Validation, X.Y., Q.X. and W.L.; Formal Analysis, Z.L.; Investigation, Q.X.; Resources, X.Z. (Xuli Zan) and L.Z.; Data Curation, D.L.; Writing—original Draft Preparation, Q.X.; Writing—review and Editing, Q.X., Q.F.; Visualization, W.L.; Supervision, X.Z. (Xiaodong Zhang) and L.D.; Project Administration, D.Z.; Funding Acquisition, X.Z. (Xiaodong Zhang). All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by [National Natural Science Foundation of China] grant number [41771104].

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data presented in this work are available on request from the corresponding author. The data are not publicly available due to other ongoing studies.

Acknowledgments: The paper is funded by the China Scholarship Council.

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

Abbreviations

The following abbreviations are used in this manuscript:

MCcGAN	Multi-channels Conditional Generative Adversarial Network
SAR	Synthetic Aperture Radar
GAN	Generative Adversarial Network
cGAN	Conditional Generative Adversarial Network
CycleGAN	Cycle-consistent Adversarial Network
MTcGAN	Multi-temporal Conditional Generative Adversarial Network
DCSTFN	Deep Convolutional Spatiotemporal Fusion Network

Appendix A

Table A1. The evaluation metrics of different methods for the red band in Area B at Time III (the bold result is the best among the different metrics).

	MCcGAN	MTcGAN	DCSTFN-OS	RDCSTFN-OS	CycleGAN	S-CycleGAN	pix2pix
RMSE	**7.5027**	8.1381	7.8189	8.1528	9.8816	8.7426	8.5784
R^2	**0.7050**	0.6529	0.6796	0.6517	0.4883	0.5995	0.6144
KGE	**0.7339**	0.5857	0.6663	0.5861	−0.5290	0.6298	0.6418
SSIM	**0.8175**	0.7560	0.7879	0.7423	0.3988	0.6504	0.6948
PSNR	**28.0365**	25.7574	26.8656	25.5598	19.7971	25.7830	25.8962

Table A2. The evaluation metrics of different methods for the green band in Area B at Time III (the bold result is the best among the different metrics).

	MCcGAN	MTcGAN	DCSTFN-OS	RDCSTFN-OS	CycleGAN	S-CycleGAN	pix2pix
RMSE	**6.5593**	7.4527	6.9871	7.3883	9.2839	7.8818	7.7580
R^2	**0.6233**	0.5137	0.5725	0.5220	0.2453	0.4561	0.4730
KGE	**0.7709**	0.6264	0.7159	0.6406	−0.4313	0.6020	0.6126
SSIM	**0.8498**	0.8096	0.8307	0.8043	0.4781	0.6924	0.7391
PSNR	**30.5869**	28.4043	29.6031	28.2778	22.6195	27.8933	28.2100

Table A3. The evaluation metrics of different methods for the blue band in Area B at Time III (the bold result is the best among the different metrics).

	MCcGAN	MTcGAN	DCSTFN-OS	RDCSTFN-OS	CycleGAN	S-CycleGAN	pix2pix
RMSE	**6.5609**	7.3773	7.0045	7.1508	8.8887	7.7986	7.8470
R^2	**0.5015**	0.3697	0.4318	0.4078	0.0851	0.2957	0.2869
KGE	**0.7144**	0.5601	0.6462	0.6026	−0.3309	0.4902	0.4943
SSIM	**0.8229**	0.7593	0.7928	0.7742	0.4720	0.6629	0.6958
PSNR	**30.7116**	28.6120	29.5962	28.9182	24.1550	28.0674	28.0847

Table A4. The evaluation metrics of different methods for the red band in Area B at Time IV (the bold result is the best among the different metrics).

	MCcGAN	MTcGAN	DCSTFN-OS	RDCSTFN-OS	CycleGAN	S-CycleGAN	pix2pix
RMSE	**8.4251**	8.7504	8.6620	8.6856	9.2409	9.0099	9.0568
R^2	**0.4846**	0.4440	0.4552	0.4522	0.3800	0.4106	0.4044
KGE	**0.5214**	0.2814	0.4315	0.2833	−0.3292	0.4188	0.4089
SSIM	**0.7503**	0.6601	0.7186	0.6485	0.4676	0.6326	0.6604
PSNR	**25.2356**	22.8284	24.3769	22.7429	21.1837	24.6746	24.2663

Table A5. The evaluation metrics of different methods for the green band in Area B at Time IV (the bold result is the best among the different metrics).

	MCcGAN	MTcGAN	DCSTFN-OS	RDCSTFN-OS	CycleGAN	S-CycleGAN	pix2pix
RMSE	**7.0170**	8.2550	7.6647	8.2568	8.5045	7.8196	7.8667
R^2	**0.4803**	0.2808	0.3800	0.2805	0.2367	0.3547	0.3469
KGE	**0.6651**	0.4720	0.5930	0.4654	−0.2499	0.4592	0.4771
SSIM	**0.8115**	0.7614	0.7892	0.7447	0.5397	0.6856	0.7254
PSNR	**29.3129**	27.0277	28.4517	26.7341	24.0982	27.7567	27.7938

Table A6. The evaluation metrics of different methods for the blue band in Area B at Time IV (the bold result is the best among the different metrics).

	MCcGAN	MTcGAN	DCSTFN-OS	RDCSTFN-OS	CycleGAN	S-CycleGAN	pix2pix
RMSE	**6.6850**	7.7798	7.2630	7.5972	8.2887	7.5881	7.6326
R^2	**0.4224**	0.2177	0.3182	0.2540	0.1120	0.2558	0.2470
KGE	**0.6161**	0.4467	0.5493	0.4976	−0.1595	0.3386	0.3649
SSIM	**0.7866**	0.7190	0.7576	0.7313	0.5247	0.6619	0.6889
PSNR	**29.9146**	27.8511	29.0041	28.2843	25.4696	28.2016	27.7938

Table A7. The time of Sentinel-1/Sentinel-2 image pairs for different areas.

Areas	S1 I	S2 I	S1 II	S2 II
Area A	28 July 2019	26 July 2019	21 August 2019	20 August 2019
Area B	26 September 2019	26 September 2019	7 November 2019	5 November 2019
Area C	14 August 2019	13 August 2019	20 September 2019	22 September 2019
Area D	20 October 2019	20 October 2019	25 October 2019	25 October 2019
Area E	12 August 2019	14 August 2019	19 August 2019	19 August 2019
Area F	15 August 2019	15 August 2019	28 August 2019	30 August 2019
Area G	5 September 2019	3 September 2019	17 September 2019	18 September 2019
Area H	6 July 2019	5 July 2019	22 September 2019	23 September 2019
Area I	4 August 2019	5 August 2019	16 August 2019	15 August 2019

Table A7. *Cont.*

Areas	S1 I	S2 I	S1 II	S2 II
Area J	27 September 2019	28 September 2019	14 October 2019	13 October 2019
Area K	10 August 2019	9 August 2019	22 August 2019	24 August 2019
Area L	19 September 2019	20 September 2019	25 September 2019	27 September 2019
Area M	14 July 2019	16 July 2019	31 July 2019	31 July 2019
Area N	1 September 2019	5 September 2019	13 September 2019	15 September 2019
Area 1	17 August 2019	16 August 2019	22 September 2019	20 September 2019
Area 2	5 August 2019	4 August 2019	18 August 2019	19 August 2019
Area 3	7 August 2019	6 August 2019	24 August 2019	26 August 2019
Area 4	2 August 2019	5 August 2019	14 August 2019	15 August 2019
Area 5	27 September 2019	23 September 2019	9 October 2019	8 October 2019

Table A8. The evaluation metrics for NDVI of different methods in different areas (the bold result is the best among the different metrics; the underlined is the second best).

		MCcGAN	MTcGAN	CycleGAN	S-CycleGAN	pix2pix
Area 1	RMSE	**0.1084**	0.1348	0.3124	0.2344	0.2149
	R^2	**0.8673**	0.7946	−0.1027	0.3791	0.4783
	KGE	**0.9168**	0.8241	0.1045	0.4465	0.5095
	SSIM	**0.7758**	0.7378	0.1960	0.2439	0.3249
	PSNR	**25.3238**	23.4240	16.1266	18.6206	19.3768
	SAM	**0.1658**	0.1869	0.4868	0.3656	0.3495
Area 2	RMSE	0.1380	**0.1159**	0.2630	0.2270	0.1892
	R^2	0.6682	**0.7662**	−0.2043	0.1029	0.3767
	KGE	0.7949	**0.8680**	0.0847	0.1789	0.4024
	SSIM	**0.7629**	0.7568	0.3210	0.3174	0.4254
	PSNR	23.2209	**24.7412**	17.6223	18.9014	20.4828
	SAM	0.2233	**0.1897**	0.4391	0.3801	0.3123
Area 3	RMSE	**0.0631**	0.0857	0.1387	0.1696	0.2214
	R^2	**0.7012**	0.4498	−0.4428	−1.1568	−2.6729
	KGE	**0.8566**	0.7614	0.2819	0.3433	0.1904
	SSIM	**0.8772**	0.8111	0.4240	0.3920	0.4748
	PSNR	**30.0163**	27.3642	23.1773	21.4313	19.1183
	SAM	**0.1353**	0.1616	0.3090	0.2829	0.2577
Area 4	RMSE	**0.0464**	0.0633	0.1391	0.1831	0.2102
	R^2	**0.9512**	0.9092	0.5618	0.2413	0.0001
	KGE	**0.9052**	0.8834	0.6442	0.5321	0.4099
	SSIM	**0.9106**	0.8679	0.5971	0.3867	0.6225
	PSNR	**32.6886**	29.9893	23.1525	20.7681	19.5693
	SAM	**0.0996**	0.1366	0.3138	0.3859	0.3514
Area 5	RMSE	0.2209	0.2514	0.2672	**0.2121**	0.2500
	R^2	0.0513	−0.2294	−0.3887	**0.1250**	−0.2149
	KGE	**0.5453**	0.4374	−0.2297	0.2079	0.3023
	SSIM	**0.6767**	0.6328	0.4341	0.3739	0.5122
	PSNR	19.1377	18.0120	17.4831	**19.4892**	18.0636
	SAM	**0.2942**	0.3124	0.4690	0.3871	0.3849

References

1. Desnos, Y.L.; Borgeaud, M.; Doherty, M.; Rast, M.; Liebig, V. The European Space Agency's Earth Observation Program. *IEEE Geosci. Remote Sens. Mag.* **2014**, *2*, 37–46. [CrossRef]
2. Ren, T.; Liu, Z.; Zhang, L.; Liu, D.; Xi, X.; Kang, Y.; Zhao, Y.; Zhang, C.; Li, S.; Zhang, X. Early Identification of Seed Maize and Common Maize Production Fields Using Sentinel-2 Images. *Remote Sens.* **2020**, *12*, 2140. [CrossRef]

3. Bontemps, S.; Arias, M.; Cara, C.; Dedieu, G.; Guzzonato, E.; Hagolle, O.; Inglada, J.; Matton, N.; Morin, D.; Popescu, R.; et al. Building a data set over 12 globally distributed sites to support the development of agriculture monitoring applications with Sentinel-2. *Remote Sens.* **2015**, *7*, 16062–16090. [CrossRef]
4. Jelínek, Z.; Mašek, J.; Starỳ, K.; Lukáš, J.; Kumhálová, J. Winter wheat, Winter Rape and Poppy Crop Growth Evaluation with the Help of Remote and Proximal Sensing Measurements. 2020. Available online: https://doi.org/10.15159/ar.20.176 (accessed on 10 August 2020).
5. Schwieder, M.; Buddeberg, M.; Kowalski, K.; Pfoch, K.; Bartsch, J.; Bach, H.; Pickert, J.; Hostert, P. Estimating Grassland Parameters from Sentinel-2: A Model Comparison Study. *PFG J. Photogramm. Remote Sens. Geoinf. Sci.* **2020**, *88*, 379–390. [CrossRef]
6. Feng, Q.; Liu, J.; Gong, J. Urban flood mapping based on unmanned aerial vehicle remote sensing and random forest classifier—A case of Yuyao, China. *Water* **2015**, *7*, 1437–1455. [CrossRef]
7. Yang, N.; Liu, D.; Feng, Q.; Xiong, Q.; Zhang, L.; Ren, T.; Zhao, Y.; Zhu, D.; Huang, J. Large-scale crop mapping based on machine learning and parallel computation with grids. *Remote Sens.* **2019**, *11*, 1500. [CrossRef]
8. Cao, R.; Chen, Y.; Chen, J.; Zhu, X.; Shen, M. Thick cloud removal in Landsat images based on autoregression of Landsat time-series data. *Remote Sens. Environ.* **2020**, *249*, 112001. [CrossRef]
9. Zhang, L.; Liu, Z.; Liu, D.; Xiong, Q.; Yang, N.; Ren, T.; Zhang, C.; Zhang, X.; Li, S. Crop Mapping Based on Historical Samples and New Training Samples Generation in Heilongjiang Province, China. *Sustainability* **2019**, *11*, 5052. [CrossRef]
10. Tan, Z.; Yue, P.; Di, L.; Tang, J. Deriving high spatiotemporal remote sensing images using deep convolutional network. *Remote Sens.* **2018**, *10*, 1066. [CrossRef]
11. Zhu, X.; Cai, F.; Tian, J.; Williams, T.K.A. Spatiotemporal fusion of multisource remote sensing data: Literature survey, taxonomy, principles, applications, and future directions. *Remote Sens.* **2018**, *10*, 527.
12. Hilker, T.; Wulder, M.A.; Coops, N.C.; Seitz, N.; White, J.C.; Gao, F.; Masek, J.G.; Stenhouse, G. Generation of dense time series synthetic Landsat data through data blending with MODIS using a spatial and temporal adaptive reflectance fusion model. *Remote Sens. Environ.* **2009**, *113*, 1988–1999. [CrossRef]
13. Weng, Q.; Fu, P.; Gao, F. Generating daily land surface temperature at Landsat resolution by fusing Landsat and MODIS data. *Remote Sens. Environ.* **2014**, *145*, 55–67. [CrossRef]
14. Feng, Q.; Yang, J.; Zhu, D.; Liu, J.; Guo, H.; Bayartungalag, B.; Li, B. Integrating multitemporal sentinel-1/2 data for coastal land cover classification using a multibranch convolutional neural network: A case of the Yellow River Delta. *Remote Sens.* **2019**, *11*, 1006. [CrossRef]
15. Wang, J.; Yang, X.; Yang, X.; Jia, L.; Fang, S. Unsupervised change detection between SAR images based on hypergraphs. *ISPRS J. Photogramm. Remote Sens.* **2020**, *164*, 61–72. [CrossRef]
16. Torres, R.; Snoeij, P.; Geudtner, D.; Bibby, D.; Davidson, M.; Attema, E.; Potin, P.; Rommen, B.; Floury, N.; Brown, M.; et al. GMES Sentinel-1 mission. *Remote Sens. Environ.* **2012**, *120*, 9–24. [CrossRef]
17. Li, Y.; Fu, R.; Meng, X.; Jin, W.; Shao, F. A SAR-to-Optical Image Translation Method Based on Conditional Generation Adversarial Network (cGAN). *IEEE Access* **2020**, *8*, 60338–60343. [CrossRef]
18. Fuentes Reyes, M.; Auer, S.; Merkle, N.; Henry, C.; Schmitt, M. Sar-to-optical image translation based on conditional generative adversarial networks—Optimization, opportunities and limits. *Remote Sens.* **2019**, *11*, 2067. [CrossRef]
19. Hinton, G.E.; Salakhutdinov, R.R. Reducing the dimensionality of data with neural networks. *Science* **2006**, *313*, 504–507. [CrossRef]
20. Wang, P.; Patel, V.M. Generating high quality visible images from SAR images using CNNs. In Proceedings of the 2018 IEEE Radar Conference (RadarConf18), Oklahoma City, OK, USA, 23–27 April 2018; pp. 570–575.
21. Feng, Q.; Yang, J.; Liu, Y.; Ou, C.; Zhu, D.; Niu, B.; Liu, J.; Li, B. Multi-Temporal Unmanned Aerial Vehicle Remote Sensing for Vegetable Mapping Using an Attention-Based Recurrent Convolutional Neural Network. *Remote Sens.* **2020**, *12*, 1668. [CrossRef]
22. Goodfellow, I.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; Bengio, Y. Generative adversarial nets. In *Advances in Neural Information Processing Systems*; MIT Press: Cambridge, MA, USA, 2014; pp. 2672–2680.
23. Isola, P.; Zhu, J.Y.; Zhou, T.; Efros, A.A. Image-to-image translation with conditional adversarial networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 1125–1134.
24. Zhu, J.Y.; Park, T.; Isola, P.; Efros, A.A. Unpaired image-to-image translation using cycle-consistent adversarial networks. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 2223–2232.
25. Fu, S.; Xu, F.; Jin, Y.Q. Reciprocal translation between SAR and optical remote sensing images with cascaded-residual adversarial networks. *arXiv* **2019**, arXiv:1901.08236.
26. Yi, Z.; Zhang, H.; Tan, P.; Gong, M. Dualgan: Unsupervised dual learning for image-to-image translation. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 2849–2857.
27. Radford, A.; Metz, L.; Chintala, S. Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv* **2015**, arXiv:1511.06434.
28. Huang, B.; Zhi, L.; Yang, C.; Sun, F.; Song, Y. Single Satellite Optical Imagery Dehazing using SAR Image Prior Based on conditional Generative Adversarial Networks. In Proceedings of the IEEE Winter Conference on Applications of Computer Vision, Snowmass Village, CO, USA, 1–5 March 2020; pp. 1806–1813.

29. Liu, L.; Lei, B. Can SAR images and optical images transfer with each other? In Proceedings of the IGARSS 2018-2018 IEEE International Geoscience and Remote Sensing Symposium, Valencia, Spain, 22–27 July 2018; pp. 7019–7022.
30. Grohnfeldt, C.; Schmitt, M.; Zhu, X. A conditional generative adversarial network to fuse sar and multispectral optical data for cloud removal from sentinel-2 images. In Proceedings of the IGARSS 2018—2018 IEEE International Geoscience and Remote Sensing Symposium, Valencia, Spain, 22–27 July 2018; pp. 1726–1729.
31. Wang, L.; Xu, X.; Yu, Y.; Yang, R.; Gui, R.; Xu, Z.; Pu, F. SAR-to-optical image translation using supervised cycle-consistent adversarial networks. *IEEE Access* **2019**, *7*, 129136–129149. [CrossRef]
32. Merkle, N.; Auer, S.; Müller, R.; Reinartz, P. Exploring the potential of conditional adversarial networks for optical and SAR image matching. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2018**, *11*, 1811–1820. [CrossRef]
33. Enomoto, K.; Sakurada, K.; Wang, W.; Kawaguchi, N.; Matsuoka, M.; Nakamura, R. Image translation between SAR and optical imagery with generative adversarial nets. In Proceedings of the IGARSS 2018-2018 IEEE International Geoscience and Remote Sensing Symposium, Valencia, Spain, 22–27 July 2018; pp. 1752–1755.
34. Li, J.; Wu, Z.; Hu, Z.; Zhang, J.; Li, M.; Mo, L.; Molinier, M. Thin cloud removal in optical remote sensing images based on generative adversarial networks and physical model of cloud distortion. *ISPRS J. Photogramm. Remote Sens.* **2020**, *166*, 373–389. [CrossRef]
35. Wang, X.; Xu, G.; Wang, Y.; Lin, D.; Li, P.; Lin, X. Thin and Thick Cloud Removal on Remote Sensing Image by Conditional Generative Adversarial Network. In Proceedings of the IGARSS 2019-2019 IEEE International Geoscience and Remote Sensing Symposium, Yokohama, Japan, 28 July–2 August 2019; pp. 1426–1429.
36. Gao, J.; Yuan, Q.; Li, J.; Zhang, H.; Su, X. Cloud Removal with Fusion of High Resolution Optical and SAR Images Using Generative Adversarial Networks. *Remote Sens.* **2020**, *12*, 191. [CrossRef]
37. Meraner, A.; Ebel, P.; Zhu, X.X.; Schmitt, M. Cloud removal in Sentinel-2 imagery using a deep residual neural network and SAR-optical data fusion. *ISPRS J. Photogramm. Remote Sens.* **2020**, *166*, 333–346. [CrossRef]
38. Gillies, D.J.; Rodgers, J.R.; Gyacskov, I.; Roy, P.; Kakani, N.; Cool, D.W.; Fenster, A. Deep Learning Segmentation of General Interventional Tools in Two-dimensional Ultrasound Images. *Med. Phys.* **2020**, *47*, 4956–4970. [CrossRef]
39. Öztürk, Ş.; Akdemir, B. HIC-net: A deep convolutional neural network model for classification of histopathological breast images. *Comput. Electr. Eng.* **2019**, *76*, 299–310. [CrossRef]
40. Bermudez, J.; Happ, P.; Oliveira, D.; Feitosa, R. Sar to optical image synthesis for cloud removal with generative adversarial networks. *ISPRS Ann. Photogramm. Remote Sens. Spat. Inf. Sci.* **2018**, *4*, 5–11. [CrossRef]
41. Mirza, M.; Osindero, S. Conditional generative adversarial nets. *arXiv* **2014**, arXiv:1411.1784.
42. Ronneberger, O.; Fischer, P.; Brox, T. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*; Springer: Munich, Germany, 2015; pp. 234–241.
43. He, W.; Yokoya, N. Multi-temporal sentinel-1 and-2 data fusion for optical image simulation. *ISPRS Int. J. Geo-Inf.* **2018**, *7*, 389. [CrossRef]
44. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.
45. Gupta, H.V.; Kling, H.; Yilmaz, K.K.; Martinez, G.F. Decomposition of the mean squared error and NSE performance criteria: Implications for improving hydrological modelling. *J. Hydrol.* **2009**, *377*, 80–91. [CrossRef]
46. Wang, Z.; Bovik, A.C.; Sheikh, H.R.; Simoncelli, E.P. Image quality assessment: From error visibility to structural similarity. *IEEE Trans. Image Process.* **2004**, *13*, 600–612. [CrossRef] [PubMed]
47. Hore, A.; Ziou, D. Image quality metrics: PSNR vs. SSIM. In Proceedings of the 2010 20th International Conference on Pattern Recognition, Istanbul, Turkey, 23–26 August 2010; pp. 2366–2369.
48. Yuhas, R.H.; Goetz, A.F.; Boardman, J.W. Discrimination among Semi-Arid Landscape Endmembers Using the Spectral Angle Mapper (SAM) Algorithm. 1992. Available online: https://core.ac.uk/download/pdf/42789956.pdf (accessed on 10 August 2020).