Residual Encoder–Decoder Conditional Generative Adversarial Network for Pansharpening

Zhimin Shao, Zexin Lu, Maosong Ran, Leyuan Fang, *Senior Member, IEEE*, Jiliu Zhou, *Senior Member, IEEE*, and Yi Zhang, *Senior Member, IEEE*

***Abstract*— Due to the limitation of the satellite sensor, it is difficult to acquire a high-resolution (HR) multispectral (HRMS) image directly. The aim of pansharpening (PNN) is to fuse the spatial in panchromatic (PAN) with the spectral information in multispectral (MS). Recently, deep learning has drawn much attention, and in the field of remote sensing, several pioneering attempts have been made related to PNN. However, the big size of remote sensing data will produce more training samples, which require a deeper neural network. Most current networks are relatively shallow and raise the possibility of detail loss. In this letter, we propose a residual encoder–decoder conditional generative adversarial network (RED-cGAN) for PNN to produce more details with sharpened images. The proposed method combines the idea of an autoencoder with generative adversarial network (GAN), which can effectively preserve the spatial and spectral information of the PAN and MS images simultaneously. First, the residual encoder–decoder module is adopted to extract the multiscale features from the last step to yield pansharpened images and relieve the training difficulty caused by deepening the network layers. Second, to further enhance the performance of the generator to preserve more spatial information, a conditional discriminator network with the input of PAN and MS images is proposed to encourage that the estimated MS images share the same distribution as that of the referenced HRMS images. The experiments conducted on the Worldview2 (WV2) and Worldview3 (WV3) images demonstrate that our proposed method provides better results than several state-of-the-art PNN methods.**

***Index Terms*— Deep learning, generative adversarial network (GAN), multispectral (MS) image, panchromatic (PAN), pansharpening (PNN).**

I. Introduction

MULTIRESOLUTION images are widespread in remote sensing, as they provide us with images at the highest resolution in both the spatial and spectral domains. However, due to the physical and hardware limitations [1], satellites often only measure a high-resolution (HR) panchromatic (PAN) image and a low-resolution multispectral (LRMS) image. The goal of PAN and multispectral (MS) image fusion [i.e., pansharpening (PNN)] is to produce a high-resolution multispectral (HRMS) image. The HRMS images are critical for subsequent tasks such as object detection and weather monitoring [2], [3].

During recent decades, various methods have been developed for PNN. Early methods focused on component substitution, and the representative methods include intensity-hue-saturation (IHS) [4], principal component analysis (PCA) [5], and Gram-Schmidt adaptive (GSA) transform [6]. Dou *et al.* [7] summarized the methods and proposed a general framework for component substitution based on the methods. These methods are straightforward and fast and can achieve high spatial resolution but are defeated by spectral distortions. Conversely, multiresolution-analysis-based methods, including Laplacian pyramid transform [8], wavelet transform [9], and support value transform, can efficiently preserve the spectral information. However, these methods may suffer from the aliasing distortion if decimation is used to decompose the PAN and LRMS images [10].

Recently, deep learning has achieved great success in various computer vision and image processing tasks [11], and several attempts have been made to apply the deep learning techniques to remote sensing image fusion. For instance, Giuseppe *et al.* [12] first proposed a lightweight three-layer convolutional neural network for PNN aided by some nonlinear radiometric indices to augment the input. After that, Scarpa *et al.* [13] further improved the performance and robustness of PNN by exploring the different variations. Yang *et al.* [14] incorporated the domain-knowledge into the deep network architecture to improve the performance. Wei *et al.* [15] introduced residual connections to make full use of the high nonlinearity of the deep learning models. Recently, the generative adversarial network (GAN) has drawn much attention due to its powerful ability to generate samples that are indistinguishable from real images [16]. GAN has been successfully introduced into many low-level vision tasks [11]. These tasks have a similar requirement for detail preservation as that of PNN. To our knowledge, the only task for PNN with GAN was proposed in [17]. The authors embedded a two-stream fusion network into the GAN framework. However, most current networks are relatively shallow and hard to fully explore the underlying power with the rapid increasing number of training samples produced by the big size of remote sensing data. Just simply increasing the number of layers will cause two problems: training difficulty and loss of details.

To deal with these problems, in this letter, we propose a residual encoder–decoder conditional generative adversarial network (RED-cGAN) for PNN. The proposed method

Manuscript received July 22, 2019; revised September 11, 2019 and October 21, 2019; accepted October 23, 2019. Date of publication November 8, 2019; date of current version August 28, 2020. This work was supported in part by the National Natural Science Foundation of China under Grant 61671312 and Grant 61922029 and in part by the Sichuan Science and Technology Program under Grant 2018HH0070. *(Corresponding authors: Leyuan Fang; Yi Zhang.)*

Z. Shao, Z. Lu, M. Ran, J. Zhou, and Y. Zhang are with the College of Computer Science, Sichuan University, Chengdu 610065, China (e-mail: yzhang@scu.edu.cn).

L. Fang is with the College of Electrical and Information Engineering, Hunan University, Changsha 410082, China (e-mail: fangleyuan@gmail.com).

This article has supplementary downloadable material available at http://ieeexplore.ieee.org, provided by the authors.

Color versions of one or more of the figures in this letter are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/LGRS.2019.2949745

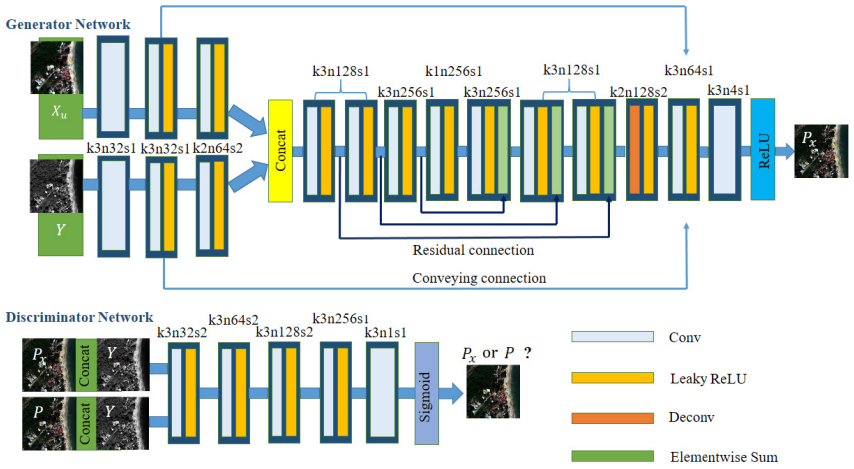

Fig. 1. Architecture of (a) generator and (b) discriminator network with the corresponding kernel size (k), number of feature maps (n), and stride (s) indicated in each convolutional layer.

combines the idea of the autoencoder [18] with GAN, which can effectively preserve the spatial and spectral information from the PAN and MS images simultaneously. The main contributions are summarized as follows: first, the residual encoder–decoder module is adopted to extract the multiscale features from different intermediate layers delivered by three residual connections to yield pansharpened images and relieve the training difficulty caused by deepening the network layers. Second, a conditional discriminator network with the input of PAN and MS images is proposed to encourage that the estimated MS images share the same distribution as that of the referenced HRMS images. It can separately extract and deliver the spectral and spatial features and efficiently measure the discrepancy between distributions.

The remainder of this letter is organized as follows. The proposed method is presented in detail in Section II, and Section III demonstrates the experimental results. Finally, the conclusions are presented in Section IV.

II. Pansharpening Method With RED-cGAN

In this letter, we assign the LRMS X the size of $W \times H \times B$, the HR PAN image Y the size of $sW \times sH \times 1$, the pansharpened image P_x, the up-sampled LRMS X_u, and the ideal HRMS images P the size of $sW \times sH \times B$, respectively. $s = 4$ is the scale factor of X and Y, and $B = 4$ indicates the number of bands of the image.

A. Pansharpening Based on cGAN

The conditional adversarial network has achieved great success in many image-to-image tasks. Different from the GAN that Goodfellow *et al.* [16] first proposed, cGAN has been proven efficient to estimate the output image conditioned on an observed image [19]. For the PNN problem, cGAN includes two parts: the generator G is used to produce P_x and the discriminator is used to distinct (Y, P) or (Y, P_x). The detailed architecture and the corresponding parameters are shown in Fig. 1. The objective function of cGAN to solve the PNN problem can be expressed as follows:

$$\mathcal{L}_{\text{cGAN}}(G, D) = \mathbb{E}_{Y\sim p_{\text{data}}(Y), P\sim p_r(P)}[\log D(Y, P)] + \mathbb{E}_{(X_u,Y)\sim p_{\text{data}}(X_u,Y)}[1 - D(Y, G(X_u, Y))] \tag{1}$$

where $D(Y, P)$ denotes the output of the conditional discriminator D, while $G(X_u, Y)$ denotes the output of the generator G. $\mathbb{E}$ represents the calculation of the expected value of the data distribution, like $p_r(P)$.

In the training step, G needs to generate images that are as close as possible to the original image, while D needs to distinguish whether the generated image is original or pansharpened. This can be seen as a min–max problem, and the objective function to solve this problem is expressed as follows:

$$G^* = \arg\min_G \max_D \mathcal{L}_{\text{cGAN}}(G, D). \tag{2}$$

It has been proven that L1 loss can better preserve the details and edges and has better robustness than other loss functions [13]. As other works did [13], [17], L1 loss is adopted to train G in this letter as

$$\mathcal{L}_{L_1}(G) = \mathbb{E}_{(X_u,Y)\sim p_{\text{data}}(X_u,Y), P\sim p_r(P)}[||P - G(X_u, Y)||_1]. \tag{3}$$

The final objective function needed to optimize is as follows:

$$G^* = \arg\min_G \max_D \mathcal{L}_{\text{cGAN}}(G, D) + \gamma\, \mathcal{L}_{L_1}(G) \tag{4}$$

where γ is the weighting parameter. After abundant tests, $\gamma = 100$ is a relatively robust choice in our experiments.

B. Residual Encoder–Decoder Generator

Normally, the PAN image is rich in the spatial information, while the MS image preserves the spectral information. To make full use of the spectral and spatial information, according to the classical framework of PNN, the proposed generator network has two components: feature extraction and feature fusion. Inspired by the work of [20], a two-branch subnetwork is used to extract the hierarchical features to capture the complementary information of the PAN and MS images. The two branches have similar architectures but different weights. Each branch consists of two convolutional layers followed by a leaky rectified linear unit (ReLU), and the parameter is set to 0.2. Next, a residual encoder–decoder architecture [21] is used to fuse the features extracted from the last two-branch subnetwork. The autoencoder was originally designed for unsupervised learning. The encoder part is used to extract the features, and the corresponding decoder part

is used to recover the original signal from the extracted features. The combined structure of the encoder and decoder is naturally suitable for image restoration. However, the classical autoencoder is not flexible enough to process images, as it is composed of fully connected layers that have a large number of parameters. In this letter, the convolution layer is adopted to replace the fully connected layer, and we construct a chain of convolutional layers and leaky ReLUs as the stacked encoders. The decoder part is integrated into our model for the recovery of structural details, which can be seen as image reconstruction from the extracted features. We also use a chain of convolutional layers to form the stacked decoders. Because the encoders and decoders should appear in a pair, the layers are symmetric in the proposed network. The encoder and decoder layers have the same kernel size to ensure that the input and output of the network match exactly.

Although the encoder–decoder structure can recover some of the details when the depth of network increases, the inverse problem becomes more ill-posed, and the cumulative errors will significantly weaken the performance of the network. In addition, when the number of layers increases, gradient diffusion appears and makes the network more difficult to train. Residual connections are introduced to address these issues. These connections will not only compensate for the details from increased layers but also avoid gradient diffusion. After the residual encoder–decoder part, a deconvolution layer is used to upsample the image to the original size followed by two convolution layers to refine the results. In the last layer, ReLU is used to ensure the output is not negative. Patch pair X_u and Y as the input of G. In our experiments on the simulated data, we followed the Wald's protocol [22] to produce X_u.

C. Conditional Discriminator

To further enhance the performance of the network, we apply a conditional discriminator network to determine whether the image is a real HRMS or pansharpened. As shown in Fig. 1(b), the discriminator network is composed of five layers, whose kernel size is 3×3. The stride of the first three layers is set to 2, and the stride for the last two layers is set to 1. Except for the last sigmoid layer, all the convolutional layers are followed by leaky ReLUs as the activation function. Different from PSGAN, (Y, P) and (Y, P_x) are used as the input of D instead of (X_u, P) and (X_u, P_x). This modification mainly lies on two considerations. First, MS + PAN is more consistent with the definition of PNN. The MS and PAN images can separately extract and deliver the spectral and spatial features, but X_u is upsampled from the original size of LRMS (usually $1/s$ of LRMS) and the spectral information may not be very accurate. Based on this fact, because Y has more accurate spatial information, we replace X_u with Y to improve the performance and avoid detail loss. Second, because Y has less bands than P, the proposed RED-cGAN can reduce the model scale to a certain degree.

III. Experimental Results and Analysis

In this section, we conduct several experiments using data from the Worldview2 (WV2) and the Worldview3 (WV3) satellites. The spatial resolution of the MS and PAN images captured by WV2 is 1.84 and 0.46 m, and the one captured by WV3 is 1.24 and 0.31 m, respectively. For our experiment, we randomly extract 7224 and 5820 patch pairs of size 128×128 without overlap for training from WV2 and WV3, respectively. All the results reported in this section are based on the test sets, which come from the same area of training patches, but are spatially separated. A total of 324 and 164 test images are randomly selected from WV2 and WV3, respectively. For Adam, we set the learning rate to 0.0001, the moment is set to 0.9, and the batch size is 8. Six state-of-the-art PNN methods, including adaptive wavelet luminance proportion (AWLP) [10], GSA [7], Brovey [9], DRPNN [15], PanNet [14], and PSGAN [17], are compared to validate the performance of our proposed method. All the implementations of the compared methods were downloaded from the links provided by the authors and we retrained all these models using the same data sets in this letter. The proposed RED-cGAN was implemented with TensorFlow. It took about 15 h to train RED-cGAN and PSGAN, and 16 and 20 h were spent on PanNet and DRPNN, respectively. All the experiments were performed on a workstation (Intel Xeon e5 2650 CPU and 256-GB RAM) equipped with a graphics processing unit card (GTX 1080 Ti).

TABLE I
Quantitative Results of the Simulated Experiments

	Method	SAM	SCC	Q4	ERGAS	Time(ms)
WV2	AWLP	7.1354	0.7710	0.7088	3.3886	140.7
	GSA	7.5488	0.7514	0.7128	3.2880	53.31
	Brovey	7.6392	0.7416	0.6447	3.6705	3.95
	DRPNN	3.6305	0.9613	0.9060	1.5604	41.16
	PANNET	4.4150	0.9436	0.8864	1.8388	40.08
	PSGAN	3.6833	0.9609	0.9020	1.5652	25.77
	RED-cGAN	**3.5993**	**0.9626**	**0.9070**	**1.5460**	26.00
WV3	AWLP	6.3022	0.7597	0.6030	4.0289	133.6
	GSA	4.5497	0.7548	0.6068	4.1451	56.19
	Brovey	6.5261	0.7451	0.5226	4.2693	4.07
	DRPNN	4.3927	0.9390	0.8461	2.3650	40.51
	PANNET	4.6084	0.9489	0.8704	2.0748	39.42
	PSGAN	4.1537	0.9522	0.8763	2.0316	25.03
	RED-cGAN	**4.0961**	**0.9605**	**0.8789**	**1.9950**	24.81

A. Evaluation at the Lower Scale

We followed the Wald's protocol [22] in the experiments. All original images are downsampled to 1/4 of the original size. The original size MS images are used as the references. The experiments in this section are performed on the downsampled images. Four widely used metrics, including the spectral angle mapper (SAM), the spatial correlation coefficient (sCC), the relative global synthesis errors (ERGAS), and Q4, are involved in evaluating the performance of different methods.

Table I shows the means of the quantitative results and computation times of different methods for two data sets. It can be seen that the deep-learning-based methods outperform other kinds of methods and the proposed RED-cGAN achieved the best scores. The computational time of RED-cGAN is similar to PSGAN and faster than all other networks. Figs. 2 and 3 show two cases cropped from the test site of WV2 and WV3, respectively. It can be observed that the results of AWLP, GSA, Brovey, and DRPNN suffer from obvious detail blurring and spectral distortions, while the results of other methods have better visual effects and are much closer to the ground truth, in terms of preserving both the spatial and spectral information. In Figs. 2 and 3, two selected regions (houses and field for Fig. 2 and an airport for Fig. 3) are

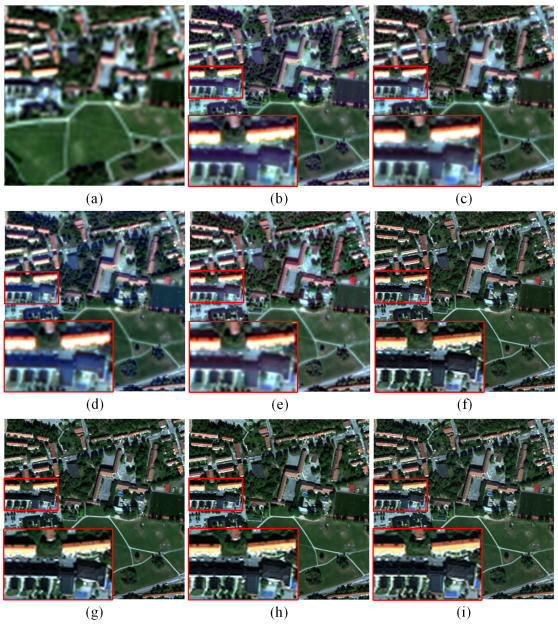

Fig. 2. Simulated fusion results from WV2. (a) Resampled LRMS. (b) AWLP. (c) GSA. (d) Brovey. (e) DRPNN. (f) PanNet. (g) PSGAN. (h) RED-cGAN. (i) Original MS.

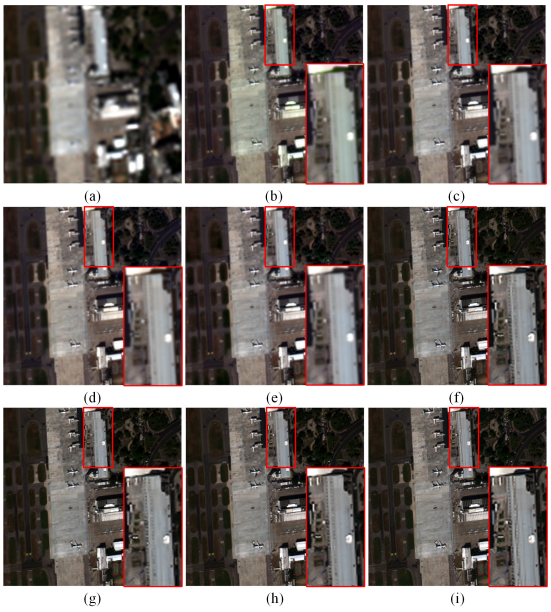

Fig. 3. Simulated fusion results from WV3. (a) Resampled LRMS. (b) AWLP. (c) GSA. (d) Brovey. (e) DRPNN. (f) PanNet. (g) PSGAN. (h) RED-cGAN. (i) Original MS.

magnified to highlight the differences among the results. From the magnified regions in Figs. 2 and 3, we can see that the spatial details are recovered more effectively by RED-cGAN than all the other methods. These merits should benefit from the proposed network structure.

B. Evaluation at the Original Scale

Different from Section III-C, all the methods are tests performed on the real data, which means that there is no ground truth. Meanwhile, the network trained with the downsampled samples (same as the Section III-C) is used.

Figs. 4 and 5 show the results for the real WV2 and WV3 data, respectively. Only qualitative results of the deep-learning-based methods are shown in effort to save space. In Fig. 4, we can see that all the methods can significantly improve the spatial resolution compared with the LRMS image; however, as demonstrated in the magnified regions, RED-cGAN produces better spatial details than other methods. The label on the building and the yellow line are clearer in Fig. 4(f). A similar trend can be observed in Fig. 5. Although the spatial structure details are improved by all the methods, RED-cGAN preserves the details better than other methods. In Fig. 5, the details for the object are the closest to that of the original PAN image as shown in Fig. 5(f). Meanwhile, spectral distortion is noticed in the image of the woods in Fig. 5(d).

To evaluate quantitatively, we use quality with no-reference (QNR) and its components D_λ (spectral component) and D_S (spatial component). The results are shown in Table II, which indicates a similar trend to the visual inspections, as the

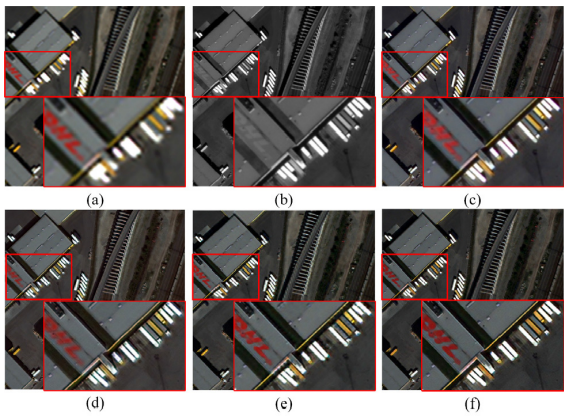

Fig. 4. Full-resolution fusion results from WV2. (a) Upsampled LRMS. (b) PAN. (c) DRPNN. (d) PanNet. (e) PSGAN. (f) RED-cGAN.

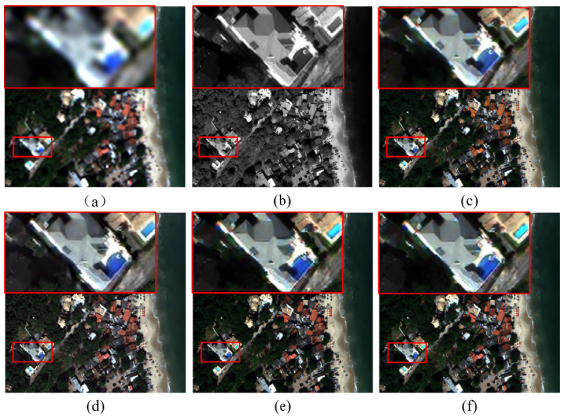

Fig. 5. Full-resolution fusion results from WV3. (a) Upsampled LRMS. (b) PAN. (c) DRPNN. (d) PanNet. (e) PSGAN. (f) RED-cGAN.

TABLE II
QUANTITATIVE RESULTS OF THE FULL-RESOLUTION EXPERIMENTS

	WV2			WV3		
	D_λ	D_s	QNR	D_λ	D_s	QNR
AWLP	0.0354	0.0361	0.9300	0.0175	0.0411	0.9421
GSA	0.0346	0.0540	0.9135	**0.0114**	0.0583	0.8647
Brovey	0.0259	0.0527	0.9185	0.0201	0.1310	0.8519
DRPNN	0.0213	0.0204	0.9590	0.0280	0.0459	0.9277
PanNet	0.0276	0.0436	0.9302	0.0377	0.0514	0.9110
PSGAN	0.0187	0.0197	0.9620	0.0180	0.0878	0.8961
RED-cGAN	**0.0180**	**0.0188**	**0.9636**	0.0130	**0.0402**	**0.9476**

TABLE III
QUANTITATIVE RESULTS OF THE FULL-RESOLUTION EXPERIMENTS

	WV2			WV3		
Methods	D_λ	D_s	QNR	D_λ	D_s	QNR
P-G	0.0175	0.0212	0.9616	0.0191	0.0749	0.9078
R-G	**0.0170**	**0.0204**	**0.9632**	**0.0140**	**0.0409**	**0.9462**
P-PAN	**0.0166**	0.0228	0.9611	0.0185	0.0906	0.8929
PSGAN	0.0187	0.0197	0.9620	0.0180	0.0908	0.8931
R-MS	0.0175	0.0200	0.9629	0.0139	0.0416	0.9457
RED-cGAN	0.0180	**0.0188**	**0.9636**	**0.0130**	**0.0402**	**0.9476**

proposed RED-cGAN achieves the best performance in most of the metrics for both data sets. It is also noticed that all the methods find that WV3 is more challenging than WV2. This can be explained by the fact WV2 and WV3 have same spectral bands but different resolution. RED-cGAN avoids this decay of performance while passing from one to another sensor and shows its better robustness.

C. Network Architecture Analysis

The differences between PSGAN and RED-cGAN lie in two parts.

1) For the generator network, we adopted multiple skip connections to transfer the multiscale features to the final results and ease the training procedure. In addition, we discarded all the pooling layers used in PSGAN, which have been proved harmful for detail preservation [18], [21].

2) For the discriminator network, the generated MS concatenated with PAN is used in our RED-cGAN.

To demonstrate the effectiveness of our appended modifications, several experiments are conducted.

1) To evaluate the improved performance by involving multiscale feature transferring and skip connections, the experiments are performed to compare the performances of the generator network of RED-cGAN (R-G) and PSGAN (P-G). In Table III, it can be seen that R-G achieves better results in terms of all the metrics in both data sets.

2) To validate the effectiveness of the proposed combined input for the discriminator network of RED-cGAN, we separately analyze the impacts of our modification on PSGAN with PAN (P-PAN) and RED-cGAN with upsampled LRMS (R-MS). Due to the page limitation, only quantitative results for full resolution experiments are shown in Table III and more results can be found in the Supplemental Materials. In Table III, with PAN and estimated MS as the input, P-PAN and RED-cGAN have higher scores than their corresponding LRMS + estimated MS versions. We also note that the modifications on the generator are fundamental for preserving the spatial details and avoid resolution loss in our experiments. Although numerical improvements are confirmed, the HR data will visually benefit more from our modification on the discriminator.

IV. CONCLUSION

In this letter, we propose a residual encoder–decoder network based on the conditional GAN framework for the PNN problem. A two-branch network is used to extract the spatial and spectral information from the PAN and MS images, respectively. Then, the RED structure uses the extracted multiscale features from the previous step to yield the pansharpened images. In addition, the discriminator network is used to distinguish whether the MS image is real or pansharpened. We evaluated the proposed method with both the simulated and real data from two different satellites. The results demonstrate that the proposed RED-cGAN outperforms other state-of-the-art methods in both qualitative and quantitative aspects.

REFERENCES

[1] W. Huang, L. Xiao, Z. Wei, H. Liu, and S. Tang, "A new pan-sharpening method with deep neural networks," *IEEE Geosci. Remote Sens. Lett.*, vol. 12, no. 5, pp. 1037–1041, May 2015.

[2] L. Zhang, L. Zhang, D. Tao, X. Huang, and B. Du, "Hyperspectral remote sensing image subpixel target detection based on supervised metric learning," *IEEE Trans. Geosci. Remote Sens.*, vol. 52, no. 8, pp. 4955–4965, Aug. 2014.

[3] C. Thomas, T. Ranchin, L. Wald, and J. Chanussot, "Synthesis of multispectral images to high spatial resolution: A critical review of fusion methods based on remote sensing physics," *IEEE Trans. Geosci. Remote Sens.*, vol. 46, no. 5, pp. 1301–1312, May 2008.

[4] S. Rahmani, M. Strait, D. Merkurjev, M. Moeller, and T. Wittman, "An adaptive IHS pan-sharpening method," *IEEE Geosci. Remote Sens. Lett.*, vol. 7, no. 4, pp. 746–750, Oct. 2010.

[5] V. P. Shah, N. H. Younan, and R. L. King, "An efficient pan-sharpening method via a combined adaptive PCA approach and contourlets," *IEEE Trans. Geosci. Remote Sens.*, vol. 46, no. 5, pp. 1323–1335, May 2008.

[6] B. Aiazzi, S. Baronti, and M. Selva, "Improving component substitution pansharpening through multivariate regression of MS+Pan data," *IEEE Trans. Geosci. Remote Sens.*, vol. 45, no. 10, pp. 3230–3239, Oct. 2007.

[7] W. Dou, Y. Chen, X. Li, and D. Z. Sui, "A general framework for component substitution image fusion: An implementation using the fast image fusion method," *Comput. Geosci.*, vol. 33, no. 2, pp. 219–228, 2007.

[8] H. Shen, X. Meng, and L. Zhang, "An integrated framework for the spatio–temporal–spectral fusion of remote sensing images," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 12, pp. 7135–7148, Dec. 2016.

[9] X. Otazu, M. González-Audícana, O. Fors, and J. Núñez, "Introduction of sensor spectral response into image fusion methods. Application to wavelet-based methods," *IEEE Trans. Geosci. Remote Sens.*, vol. 43, no. 10, pp. 2376–2385, Oct. 2005.

[10] Y. Kim, C. Lee, D. Han, Y. Kim, and Y. Kim, "Improved additive-wavelet image fusion," *IEEE Geosci. Remote Sens. Lett.*, vol. 8, no. 2, pp. 263–267, Mar. 2011.

[11] 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. 2015.

[12] G. Masi, D. Cozzolino, L. Verdoliva, and G. Scarpa, "Pansharpening by convolutional neural networks," *Remote Sens.*, vol. 8, no. 7, p. 594, Jul. 2016.

[13] G. Scarpa, S. Vitale, and D. Cozzolino, "Target-adaptive CNN-based pansharpening," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 9, pp. 5443–5457, Sep. 2018.

[14] J. Yang, X. Fu, Y. Hu, Y. Huang, X. Ding, and J. Paisley, "PanNet: A deep network architecture for pan-sharpening," in *Proc. ICCV*, Oct. 2017, pp. 5449–5457.

[15] Y. Wei, Q. Yuan, H. Shen, and L. Zhang, "Boosting the accuracy of multispectral image pansharpening by learning a deep residual network," *IEEE Geosci. Remote Sens. Lett.*, vol. 14, no. 10, pp. 1795–1799, Oct. 2017.

[16] I. Goodfellow *et al.*, "Generative adversarial nets," in *Proc. NIPS*, 2014, pp. 2672–2680.

[17] X. Liu, Y. Wang, and Q. Liu, "PSGAN: A generative adversarial network for remote sensing image pan-sharpening," in *Proc. ICIP*, Oct. 2018, pp. 873–877.

[18] J. Masci, U. Meier, D. Cireşan, and J. Schmidhuber, "Stacked convolutional auto-encoders for hierarchical feature extraction," in *Artificial Neural Networks and Machine Learning*. Espoo, Finland: Springer-Verlag, 2011.

[19] M. Mirza and S. Osindero, "Conditional generative adversarial nets," 2014, *arXiv:1411.1784*. [Online]. Available: https://arxiv.org/abs/1411.1784

[20] Z. Shao and J. Cai, "Remote sensing image fusion with deep convolutional neural network," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 11, no. 5, pp. 1656–1669, May 2018.

[21] H. Chen *et al.*, "Low-dose CT with a residual encoder-decoder convolutional neural network," *IEEE Trans. Med. Imag.*, vol. 36, no. 12, pp. 2524–2535, Dec. 2017.

[22] L. Wald, T. Ranchin, and M. Mangolini, "Fusion of satellite images of different spatial resolutions: Assessing the quality of resulting images," *Photogramm. Eng. Remote Sens.*, vol. 63, no. 6, pp. 691–699, 1997.