Information Fusion 78 (2022) 158–170

Contents lists available at ScienceDirect

Information Fusion

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

Full length article

Laplacian pyramid networks: A new approach for multispectral pansharpening

Cheng Jin [a], Liang-Jian Deng [b,*], Ting-Zhu Huang [b], Gemine Vivone [c]

[a] School of Optoelectronic Science and Engineering, University of Electronic Science and Technology of China, Chengdu, 611731, China
[b] School of Mathematical Sciences, University of Electronic Science and Technology of China, Chengdu, 611731, China
[c] Institute of Methodologies for Environmental Analysis, CNR-IMAA, 85050 Tito Scalo, Italy

ARTICLE INFO

Keywords:
Laplacian pyramid
Modulation transfer function
Convolutional neural network
Pansharpening
Image fusion
Machine learning
Remote sensing

ABSTRACT

Pansharpening is about fusing a high spatial resolution panchromatic image with a simultaneously acquired multispectral image with lower spatial resolution. In this paper, we propose a Laplacian pyramid pansharpening network architecture for accurately fusing a high spatial resolution panchromatic image and a low spatial resolution multispectral image, aiming at getting a higher spatial resolution multispectral image. The proposed architecture considers three aspects. First, we use the Laplacian pyramid method whose blur kernels are designed according to the sensors' modulation transfer functions to separate the images into multiple scales for fully exploiting the crucial spatial information at different spatial scales. Second, we develop a fusion convolutional neural network (FCNN) for each scale, combining them to form the final multi-scale network architecture. Specifically, we use recursive layers for the FCNN to share parameters across and within pyramid levels, thus significantly reducing the network parameters. Third, a total loss consisting of multiple across-scale loss functions is employed for training, yielding higher accuracy. Extensive experimental results based on quantitative and qualitative assessments exploiting benchmarking datasets demonstrate that the proposed architecture outperforms state-of-the-art pansharpening methods. Code is available at https://github.com/ChengJin-git/LPPN.

1. Introduction

Thanks to a wide range of applications, such as medicine [1], sociology [2], ecology [3], and other fields, pansharpening has drawn loads of attention from the scientific community. This can be corroborated by the organization of the data fusion contest from the IEEE Geoscience and Remote Sensing Society in 2006 [4] and a huge recent literature about this topic [5–7]. Pansharpening is about merging a high resolution panchromatic (PAN) and a low resolution multispectral (LRMS) images, which can be easily obtained by several satellite sensors like WorldView-3, QuickBird, and GaoFen. As illustrated in Fig. 1, the goal of pansharpening is to yield a high resolution multispectral (HRMS) image. Besides, pansharpening algorithms have gained much interest from commercial enterprises. For instance, famous software, such as ENVI and ERDAS, utilizes fusion techniques to deal with image enhancement problems [8]. Moreover, pansharpening has been considered a preliminary step for several image processing tasks, e.g., change detection [9], which demonstrates the important role of pansharpening in practical applications.

Regarding pansharpening algorithms, we may categorize them into four classes, i.e., component substitution (CS) approaches, multi-resolution analysis (MRA) methods, variational optimization-based (VO) approaches, and deep learning-based (DL) techniques [7]. In this work, we mainly focus on developing a new deep learning network to address the pansharpening problem.

The first two categories are CS and MRA methods which obtain the final pansharpened outcome from the perspective of details injection, see, e.g., [10,12–15]. The former approach is based on the substitution of a component of the spectral transformed multispectral (MS) image with the PAN image, see, e.g., the band-dependent spatial-detail with local parameter estimation (BDSD) [10], the robust band-dependent spatial-detail (BDSD-PC) [16], and the PRACS approach [12]. The MRA class relies upon the injection of the spatial details of the PAN image into the upsampled LRMS image to obtain a high spatial resolution MS image, see, e.g., the smoothing filter-based intensity modulation (SFIM) [17], the generalized Laplacian pyramid (GLP) [18–20], the GLP with full-scale regression (GLP-Reg) [21], the deconvolution based

* Corresponding author.
E-mail addresses: Cheng.Jin@std.uestc.edu.cn (C. Jin), liangjian.deng@uestc.edu.cn (L.-J. Deng), tingzhuhuang@126.com (T.-Z. Huang), gemine.vivone@gmail.com (G. Vivone).

https://doi.org/10.1016/j.inffus.2021.09.002
Received 11 January 2021; Received in revised form 9 July 2021; Accepted 15 September 2021
Available online 30 September 2021
1566-2535/© 2021 Elsevier B.V. All rights reserved.

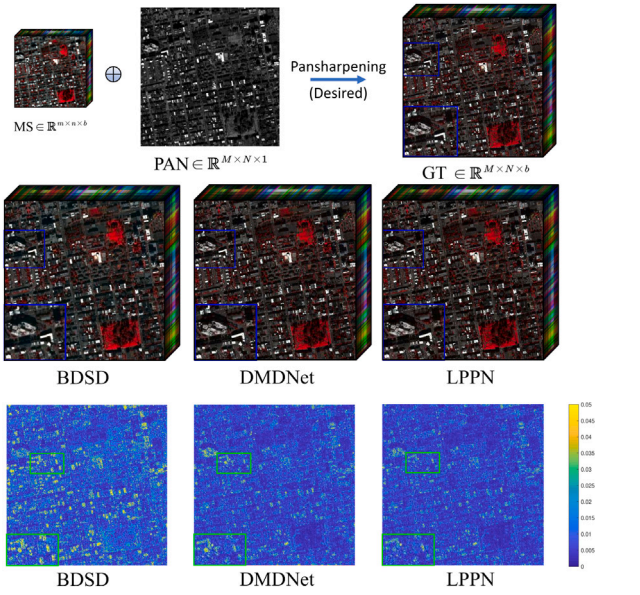

Fig. 1. First row: the schematic diagram of pansharpening, where MS $\in \mathbb{R}^{m\times n\times b}$, PAN $\in \mathbb{R}^{M\times N\times 1}$ and GT $\in \mathbb{R}^{M\times N\times b}$, $M = 4\ m, N = 4n$, b is the number of bands of the MS image. Second row: the pansharpened results by a state-of-the-art traditional method, BDSD [10], a deep learning approach, DMDNet [11], and the proposed method. Third row: the residual maps between BDSD, DMDNet, the proposed approach, and the ground-truth (GT). It is clear that our method obtains a better residual map, thus showing a better visual quality.

methods [22,23], and the bilinear filtering [24]. Hybrid CS/MRA approaches can also be found in the related literature [14,25]. In general, the two types of methods could get quite fast computation and promising outcomes, but, sometimes, may lead to slight spatial and spectral distortions.

The third category, the so-called VO class, is based on the formulation of models with proper regularizers to address the pansharpening problem, then reconstructing the high-resolution outcomes by designing algorithms for the given models. The whole process can be seen from a mathematical point of view as the reconstruction of incomplete complementary observations of multi-channel data. In general, it mainly contains Bayesian-, Sparse Reconstruction (SR)-, and Model-based Optimization (MBO) techniques, whose generally utilizes the regularization-based technique to address this challenge, see, e.g., [26–32]. In [33], Ballester et al. proposed the P+XS method, which obtained the spectral information for the fused image under the assumption that the PAN image can be approximated as a linear combination of the high resolution multispectral bands. In [34], Yokoya et al. employed the coupled nonnegative matrix factorization (CNMF) unmixing to deal with the pansharpening problem, which could produce high quality fusion results, both spectrally and spatially. In [35], Deng et al. proposed a reproducible kernel Hilbert space and Heaviside based framework to address the task of pansharpening. Recently, Wu et al. in [36] proposed a new DL-based VO scheme that could benefit from both traditional VO methods and the outcome of DL-based approaches, thus getting both competitive results and generalization ability. Especially, some recent hyperspectral and multispectral image fusion approaches can be also applied to the task of pansharpening, see e.g., [37,38]. More recently, this field of pansharpening has evolved into more specific applications, i.e., in cloud-contaminated circumstances [39], where X. Meng et al. proposed a variational-based integrated pansharpening model specifying in cloud contamination scenarios. Although VO-based techniques can obtain competitive results, they are sometimes limited, e.g., the use of many hyperparameters, the insufficient feature representation and extraction that could result in spatial and spectral distortions.

The fourth category is represented by the DL-based methods, which are mainly based on a feature extraction/representation phase, the training of network parameters, and the testing on real-world data. These methods are often far from a theoretical development and require enormous datasets to train the underlying parameters, but they have been widespread used (including but not limited to image super-resolution, pattern recognition, and image classification), often showing state-of-the-art performance. In particular, many satisfactory results have been obtained by deep learning networks for pansharpening [40–44]. In [45] , Masi et al. employed first a convolutional neural network (CNN) with three layers to deal with the task of multispectral image pansharpening, obtaining excellent pansharpened results comparing with traditional state-of-the-art approaches. In [46], Fu et al. developed a deep CNN architecture with a high-pass filtering technique to fuse the PAN and the LRMS images, which obtains state-of-the-art pansharpening outcomes. Besides, He et al. [47] followed physical concepts to design a CNN based method for pansharpening relied upon the spatial detail injection framework. In [11], Fu et al. elaborated their high-pass filtering network design with a grouped multiscale network structure. In [48], Deng et al. proposed two network architectures, named CS-Net and MRA-Net, respectively, according to the traditional CS and MRA fusion equations. To limit the drawbacks of the two networks, they further proposed a simple but effective fusion network, called FusionNet, to yield state-of-the-art pansharpening outcomes. More recently, Generative Adversarial Networks (GAN) and unsupervised training have teemed with the pansharpening field thanks to their novelty and powerfulness. Literature like [49,50] demonstrates the promising features of these method. For improving the effectiveness of these kinds of methods, the solutions mainly focus on two broad aspects. One is to make more generalized datasets for network training, e.g. [51]. The other mainly focuses on the amelioration of the network architecture, i.e., by combining the aforementioned methods [52], utilizing new state-of-the-art modules [53] and amalgamating mathematical knowledge into neural network practice [54].

Although these methods have achieved excellent performance, there is still room for improvement by considering the following points. First of all, the multi-scale property plays an important role in resolution enhancement applications, see, e.g., the image super-resolution issue [55]. It is important to design more effective multi-scale network architectures based on widely used multi-scale structures, see, e.g., the state-of-the-art Laplacian pyramids. Second, for the specific pansharpening problem, the Modulation Transfer Function (MTF) is a reasonable way to describe the physical procedure of image capture, thus its consideration in the datasets simulation is expected to get better performance.

Motivated by the above-mentioned points, we propose, in this work, a new network architecture that is able to deal with these aspects. Hence, a novel Laplacian pyramid pansharpening network (LPPN) architecture is considered. The given network accounts for both the image multi-scale information and the sensors' MTFs. To exploit the multi-scale information, we use the Laplacian pyramid to decompose the original PAN image and the upsampled LRMS image in several scales, then designing the corresponding sub-networks for each image scale and incorporating them into a fusion convolutional neural network (FCNN). Due to the use of multi-scale sub-networks, the final loss function is represented by a combination of multiple ℓ_2 loss functions. Moreover, we also introduce recursive blocks into our sub-networks, aiming to reduce the number of network parameters. Additionally, unlike the classical Laplacian pyramid that uses a fixed Gaussian kernel for all the image channels, our approach employs the specific sensors' MTFs, which vary along the spectral dimension (i.e., different Gaussian kernels for different spectral bands). Extensive experiments on reduced and full resolution datasets demonstrate that the proposed method gets the best quantitative and qualitative performance compared with state-of-the-art pansharpening approaches.

The contributions of this paper can be summarized as follows:

- To the best of our knowledge, this is the first work merging both the recursive sub-network and Laplacian pyramids to address the pansharpening task. Unlike the classical Laplacian pyramid using a fixed Gaussian kernel for all the spectral channels, the Laplacian pyramid used in our work takes advantage of the specific sensors' MTFs for image scale decomposition, which can yield significant improvements.
- The exploitation of Laplacian pyramids allow us to develop multi-scale structured sub-networks improving the capability of managing different spatial details at the feature extraction stage. Besides, we also build multiple loss functions to describe the information loss for each scale, which can aid the image details recovery at different scales.
- The recursive block is used within each sub-network to effectively decrease the number of network parameters. Furthermore, its use can increase the depth of the sub-networks. Thus, the proposed approach can be seen as a lightweight network for pansharpening. For example, our method only involves about 50,000 network parameters to achieve the state-of-the-art performance on WorldView-3 datasets, which is significantly far away from the compared approaches.

The remaining of the paper is as follows. Firstly, the related works and motivations under the development of the proposed method are reported in Section 2. Section 3 is devoted to the presentation of the proposed method, including the network architecture, the loss function, the training details, and so forth. Section 4 shows the quantitative and qualitative outcomes. Finally, conclusions are drawn in Section 5.

2. Related works and motivations

Laplacian pyramid decomposition (LPD) and its multi-scale structure, which are suitable for image resolution enhancement tasks. Recently, some pansharpening algorithms are proposed from the perspective of constructing multi-scale structures, see, e.g., [56], obtaining promising outcomes. For instance, Yuan et al. [57] formulated a novel pansharpening approach by employing the multi-scale convolutional kernels, which could fully widen the network and effectively extract the image features on different scales. In [58], although the authors mentioned the words "Laplacian pyramid" in their work, they did not really exploit the multi-scale LPD for the task of pansharpening, which limited the final pansharpening performance. Different from the introduced multi-scale methods, here we accordingly resort to the classical LPD with some special operations in pansharpening, e.g., LPD with MTF, to formulate an effective and lightweight deep CNN with a multi-scale structure, aiming to yield state-of-the-art pansharpening results. In this section, we will introduce first the Gaussian pyramid decomposition (GPD) and the related LPD. Afterwards, we will point out the motivations under the development of the proposed method.

2.1. GPD and LPD

Proposed by H. Olkkonen et al. [59], the GPD is about the reconstruction of the image by the Gaussian expand wavelet transform for multi-resolution analysis of images. It aims to use Gaussian kernels to create a series of images at different scales. The general GPD equation is as follows:

$$\begin{aligned} &\boldsymbol{G}_1(\boldsymbol{I}) = \boldsymbol{I}, \\ &\boldsymbol{G}_i(\boldsymbol{I}) = (\boldsymbol{k} \otimes \boldsymbol{G}_{i-1}(\boldsymbol{I})) \downarrow^2, i = 2, \dots, S, \end{aligned} \tag{1}$$

where $\boldsymbol{G}_i$ stands for the ith layer of the Gaussian pyramid, $\boldsymbol{I}$ is the original image, $\otimes$ denotes the convolution operation, $\downarrow^2$ indicates the downsampling with a scale factor of 2, $\boldsymbol{k}$ represents the Gaussian kernel, and S is the total number of layers. It is worth to be remarked that $\boldsymbol{k}$ is fixed for each channel of an MS image.

The LPD is a bandpass image decomposition derived from the GPD. It is originally proposed by Burt and Adelson [60] before multiresolution wavelet analysis was introduced. More in detail, the LPD technique is a multi-resolution image representation obtained through a recursive reduction of the set of data.

The general LPD processing equation is as follows:

$$\begin{aligned} &\boldsymbol{L}_S(\boldsymbol{I}) = \boldsymbol{G}_S(\boldsymbol{I}), \\ &\boldsymbol{L}_i(\boldsymbol{I}) = \boldsymbol{G}_i(\boldsymbol{I}) - \boldsymbol{G}_{i+1}(\boldsymbol{I}) \uparrow^2, i = S-1, \dots, 1, \end{aligned} \tag{2}$$

where $\boldsymbol{L}_i$ stands for the ith layer of the LPD, $\boldsymbol{G}_i$ is the outcome of the ith layer of the GPD, and $\uparrow^2$ indicates the upsampling with a scale factor of 2. Please, refer to Fig. 2 for a better understanding.

2.2. Motivations

Since conventional deep learning networks do not often exploit multi-scale information, the proposed work is motivated by the introduction of this kind of information. In particular, the pyramid structure achieves a multi-scale representation in image processing to portray global and local information in a better way. This property inspired us to utilize the multi-scale spatial representation structure of the LPD to design an end-to-end deep neural network structure capable of progressively restoring more image information to get a better performance.

It is important to note that conventional GPD methods mainly utilize a fixed Gaussian kernel for each channel of an MS image (e.g., an RGB image) to proceed with the convolution operation. However, this approach fails to meet the physics principles of pansharpening since the MS sensors' point spread functions, which model the spatial responses of each spectral channel (also called band), are not the same along the spectral dimension. In other words, the Gaussian kernels related to the spectral bands of an MS image should be different from each other. They are about a special function, i.e., the MTF. Thus, we exploit the MTF to generate the specific kernels for every band of the MS image in order to implement the GPD. For more details, the interesting readers can refer to Section 3.1.1.

3. The proposed method

In this section, we describe the design of the proposed LPPN. The whole procedure is depicted in Fig. 3. The architecture of the network, the adopted loss function, and the training details are introduced in the remaining of this section. For simplicity, we denote the PAN image as $\boldsymbol{P}$ and the MS image as $\boldsymbol{M}$.

3.1. Network architecture

This section is devoted to the detailed description of the proposed network architecture.

3.1.1. GPD with MTF

The Modulation Transfer Function (MTF) is used to model the magnitude response of the optical system at different spatial frequencies. In pansharpening, it has widespread used to design filters for image degradation. The sensors' MTFs are usually band-dependent, thus generating filters that have different Gaussian kernels for each spectral band. In this work, our goal is to address a resolution enhancement issue involving natural images, thus motivating us to implement Laplacian pyramids using the MTF *a priori* information. For more details, the interesting readers can refer to [6,61].

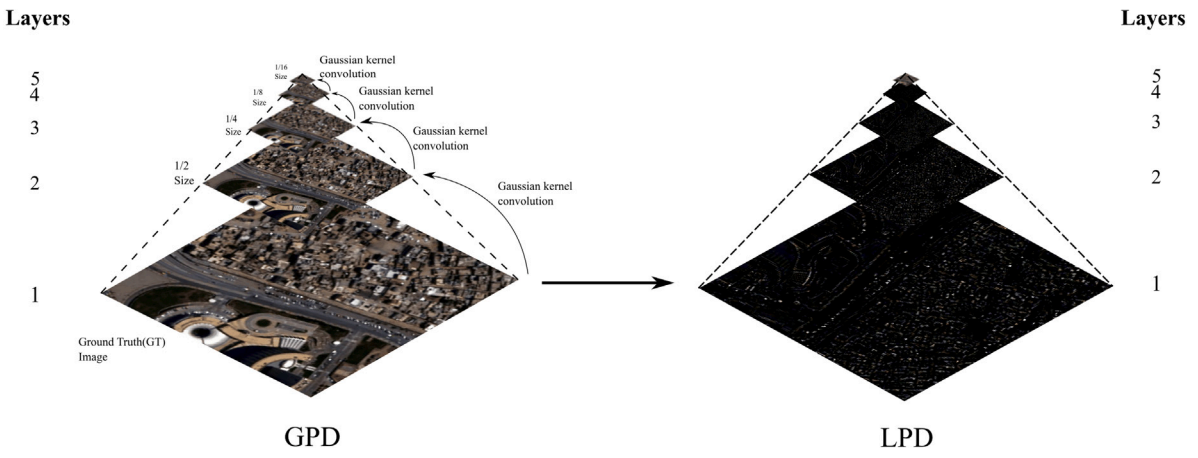

Fig. 2. Schematic diagram of the Gaussian pyramid decomposition (GPD) and the Laplacian pyramid decomposition (LPD) for $S = 5$.

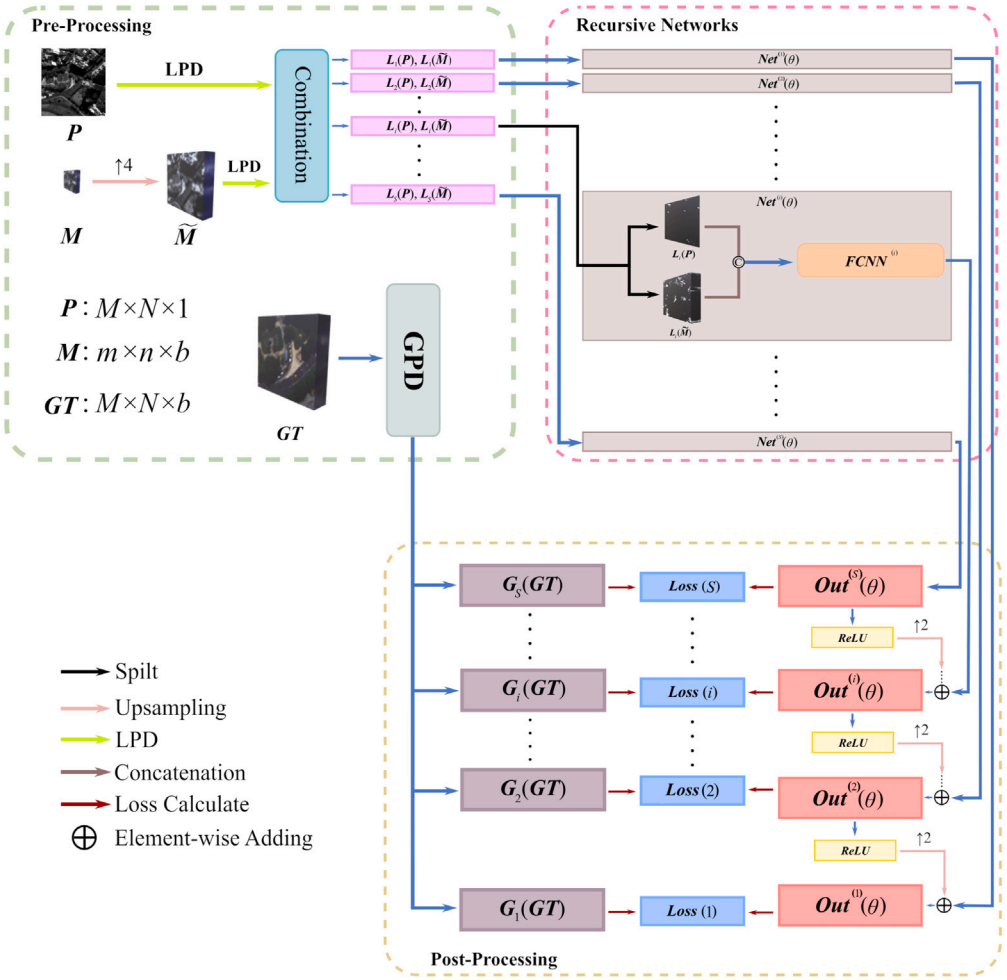

Fig. 3. The network architecture of the proposed Laplacian pyramid pansharpening network (LPPN) for a resolution ratio between PAN and MS equal to 4.

3.1.2. LPD with MTF

Multi-scale information is essential for image resolution enhancement applications. Thus, we want to utilize the LPD, a well-established technique for producing multi-scale images, to recover more image details. In what follows, we will introduce our MTF LPD, which performs first the MTF GPD into several scales of the original PAN image, the upsampled LRMS image, and the ground-truth (GT) multispectral image, thus getting the corresponding LPD components using (2).

To match the spatial resolution, we employ the transposed convolution to upsample $\boldsymbol{M}$ with a factor 4 (denoted as $\widetilde{\mathbf{M}}$) to the PAN, $\boldsymbol{P}$, scale. Then, we separately decompose $\boldsymbol{P}$ and $\widetilde{\mathbf{M}}$ into their S-layer pyramids, which is processed by the following procedure:

(a) We generate convolution kernels according to the MTF information. $\boldsymbol{P}$ and $\widetilde{\mathbf{M}}$ have corresponding MTF functions (readers can find the kernels by having a look at the codes MTF-PAN and MTF,

which are available in the pansharpening toolbox[1]). In particular, the MTF kernel for $\boldsymbol{P}$ is denoted as $\boldsymbol{k_P}$. Regarding to the bth band of $\widetilde{\mathbf{M}}$, we separately indicate each band's kernel as $\boldsymbol{k}_1, \boldsymbol{k}_2, \ldots, \boldsymbol{k}_b$, respectively.

(b) Compute the GPD with MTF for $\boldsymbol{P}$ and $\widetilde{\mathbf{M}}$. Let $\boldsymbol{G}_i(\boldsymbol{P})$, $\boldsymbol{G}_i(\widetilde{\mathbf{M}})$ denote the MTF-decomposed PAN and upsampled LRMS images of the ith Gaussian pyramid layer ($i = 1, 2, \ldots, S$), and set $\boldsymbol{G}_1(\boldsymbol{P}) = \boldsymbol{P}$, $\boldsymbol{G}_1(\widetilde{\mathbf{M}}) = \widetilde{\mathbf{M}}$. Afterwards, we respectively compute the output image at each GPD layer for $\boldsymbol{P}$

$$\begin{aligned} &\boldsymbol{G}_1(\boldsymbol{P}) = \boldsymbol{P}, \\ &\boldsymbol{G}_i(\boldsymbol{P}) = (\boldsymbol{k_P} \otimes \boldsymbol{G}_{i-1}(\boldsymbol{P})) \downarrow^2, i = 2, \ldots, S, \end{aligned} \tag{3}$$

and for $\widetilde{\mathbf{M}}$

$$\begin{aligned} &\boldsymbol{G}_1(\widetilde{\mathbf{M}}) = \widetilde{\mathbf{M}}, \\ &\boldsymbol{G}_i(\widetilde{\mathbf{M}}_j) = (\boldsymbol{k}_j \otimes \boldsymbol{G}_{i-1}(\widetilde{\mathbf{M}}_j)) \downarrow^2, i = 2, \ldots, S. \end{aligned} \tag{4}$$

where $\otimes$ denotes the convolution operation, $\downarrow^2$ stands for the downsampling operation with a scale factor of 2, and $j = 1, 2, \ldots, b$ is the band index.

(c) Compute LPD with MTF for $\boldsymbol{P}$ and $\widetilde{\mathbf{M}}$ based on the obtained GPD in (3) and (4), i.e.

$$\begin{aligned} &\boldsymbol{L}_S(\boldsymbol{P}) = \boldsymbol{G}_S(\boldsymbol{P}), \\ &\boldsymbol{L}_i(\boldsymbol{P}) = \boldsymbol{G}_i(\boldsymbol{P}) - \boldsymbol{G}_{i+1}(\boldsymbol{P}) \uparrow^2, i = S-1, \ldots, 1, \end{aligned} \tag{5}$$

and

$$\begin{aligned} &\boldsymbol{L}_S(\widetilde{\mathbf{M}}) = \boldsymbol{G}_S(\widetilde{\mathbf{M}}), \\ &\boldsymbol{L}_i(\widetilde{\mathbf{M}}_j) = \boldsymbol{G}_i(\widetilde{\mathbf{M}}_j) - \boldsymbol{G}_{i+1}(\widetilde{\mathbf{M}}_j) \uparrow^2, i = S-1, \ldots, 1, \end{aligned} \tag{6}$$

where $\uparrow^2$ stands for the upsampling operation with a scale factor of 2, and $j = 1, 2, \ldots, b$ is the band index.

More details about Laplacian pyramids can be found in [60].

In summary, the whole procedure of generating of our LPD with MTF for $\boldsymbol{P}$ and $\widetilde{\mathbf{M}}$ can be found in (5) and (6). Please, refer also to Fig. 4 for the LPD and the GPD for $\boldsymbol{P}$ and $\widetilde{\mathbf{M}}$.

3.1.3. Fusion Convolutional Neural Network (FCNN)

After getting the multi-scale LPD components of $\boldsymbol{P}$ and $\widetilde{\mathbf{M}}$, we consider them into the designed fusion convolutional neural network (FCNN, denoted as $\boldsymbol{Net}^{(i)}(\theta)$ in Fig. 3) for each scale to extract features. Thus, our final architecture is the combination of multiple sub-networks.

More in detail, we concatenate first the $\boldsymbol{L}_i(\boldsymbol{P})$ and $\boldsymbol{L}_i(\widetilde{\mathbf{M}})$ at the ith layer of LPD, after taking them into account at the corresponding sub-network $\boldsymbol{Net}^{(i)}(\theta)$ and producing the output. Furthermore, the output is operated by the ReLU activation function and upsampled exploiting a factor of 2. Finally, it is transmitted to the output of the next layer.

For each sub-network $\boldsymbol{Net}^{(i)}(\theta)$ (i.e., each FCNN), we mainly employ the residual learning [62] and recursive blocks [63] for enhancing the performance of the network, which separately play the role of improving accuracy and reducing the number of parameters. Please, see Fig. 5 for the structure of the FCNN. More details on the $\boldsymbol{Net}^{(i)}(\theta)$ sub-network are pointed out as follows.

(a) *Feature Extraction:* The goal of this step is to acquire the feature maps of the corresponding inputs. For higher layers, we take fewer kernels. In this work, we apply 2^{S-i+1} kernels with a 3×3 size for convolution operations to the concatenated input.

(b) *Recursive Blocks:* To achieve the parameter reduction and improve the accuracy, we employ several recursive blocks. As the gradient vanishing and exploding in training of deep models, the shared-source skip-connection residual learning is incorporated into our sub-network, whose accuracy is proven in [55]. The goal of our recursive block setting is to share the network parameters across each recursive block and between each convolutional layer. Therefore, a single set of parameters is capable of building multi-layer sub-networks.

(c) *Output Reconstruction:* After recursive blocks, the FCNN output for each layer is integrated to reconstruct the multiple outputs, which lead to multiple across-scale loss functions for training. In the testing phase, we get multiple outputs and the sole output at the first layer, i.e., $\mathbf{Out}^{(1)}(\theta)$, represents the outcome of the proposed approach.

Fig. 3 sums up the whole proposed architecture. It mainly includes several sub-networks to progressively fuse and enhance the spatial details. Please, see Fig. 6 to get more insights about how the proposed network architecture works and its effectiveness on an exemplary real-world test case.

3.2. Loss function

Due to the usage of the LPD, we will have multi-scale output images. For each image, we build one FCNN to train the network parameters. Therefore, the final loss function is the combination of the different loss functions. The final loss function is defined as follows:

$$\mathcal{L}(\theta^{(i)}) = \sum_{i=1}^{S} \|\mathbf{Out}^{(i)}(\theta) - \boldsymbol{G}_i(\boldsymbol{GT})\|_F^2, \tag{7}$$

where $\mathbf{Out}^{(i)}(\theta)$ is obtained by $\mathcal{F}_{\theta^{(i)}}\left(\boldsymbol{L}_i(\boldsymbol{P}), \boldsymbol{L}_i(\widetilde{\mathbf{M}})\right)$ and its subsequent operations (please see Fig. 3 for more details). In particular, i indicates the ith layer of LPD and θ stands for the network parameters. Besides, $\boldsymbol{G}_i(\boldsymbol{GT})$ is the ith layer of the GT image obtained via the GPD and $\|\cdot\|_F$ is the Frobenius norm. In this work, we empirically set S equal to 5. It is worth to be pointed out that in order to have a robust layer-by-layer injection procedure, the use of multiple losses to constrain the distance between the output image of a layer and the corresponding target (GT) image is advisable and commonly used in the related literature. If the loss function is only about the layer 1 (i.e., the final layer), the control of each layer of the network could be not easy, making the training a hard task. Moreover, although all the loss functions in (7) have equal weights, the more the training time the higher the impact of the first layer on the final solution.

3.3. Generation of training data

For the generation of training samples, we simulated three datasets acquired from the WorldView-3, the QuickBird, and the Gaofen-2 sensors. The simulation way for the three datasets is the same. We consider here the WorldView-3 datasets as an exemplary case. The WorldView-3 satellite datasets can be freely downloaded.[2] The same way as in [48] is exploited to simulate the training/validation/testing datasets getting 12,580 PAN/LRMS/GT patch pairs of size 64×64, $16 \times 16 \times 8$, and $64 \times 64 \times 8$, respectively. After that, these datasets are divided into the 70/20/10% for training (8,806 examples)/validation (2,516 examples)/testing (1,258 examples). We simulate the LRMS, the PAN, and the GT images according to Wald's protocol [64] due to the unavailability of the GT image. The upsampled LRMS image is obtained via a polynomial kernel with 23 coefficients [18], called EXP from hereon. Please, find more details on the implementation of Wald's protocol in [48].

[1] http://openremotesensing.net/knowledgebase/a-critical-comparison-among-pansharpening-algorithms/.

[2] https://www.maxar.com/product-samples/.

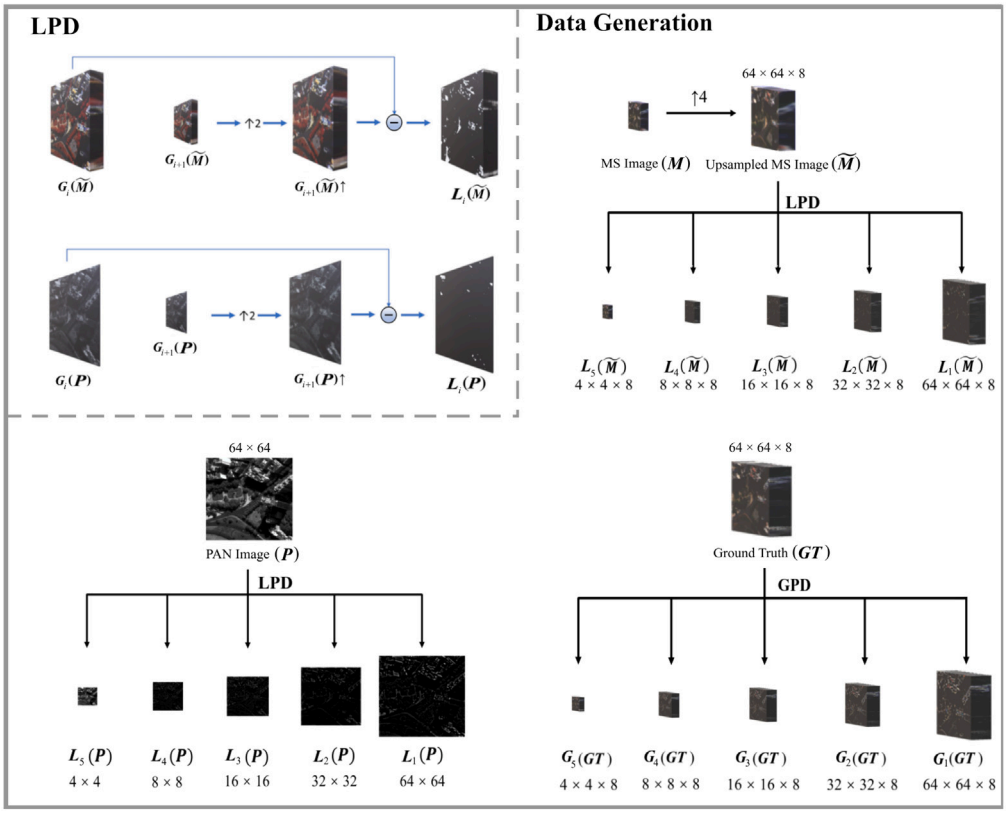

Fig. 4. Detailed illustration of the generation of the i-th Laplacian pyramid layer of $\widetilde{\mathbf{M}}$ and $\boldsymbol{P}$ (exploiting an exemplary WorldView-3 data with $S = 5$ and resolution ratio between the products to be fused equal to 4). $\boldsymbol{G}_i$ represents the i-th layer of the Gaussian pyramid. Note that we do not use the classical Gaussian pyramid, rather we use MTF-based filters to generate Laplacian pyramid layers.

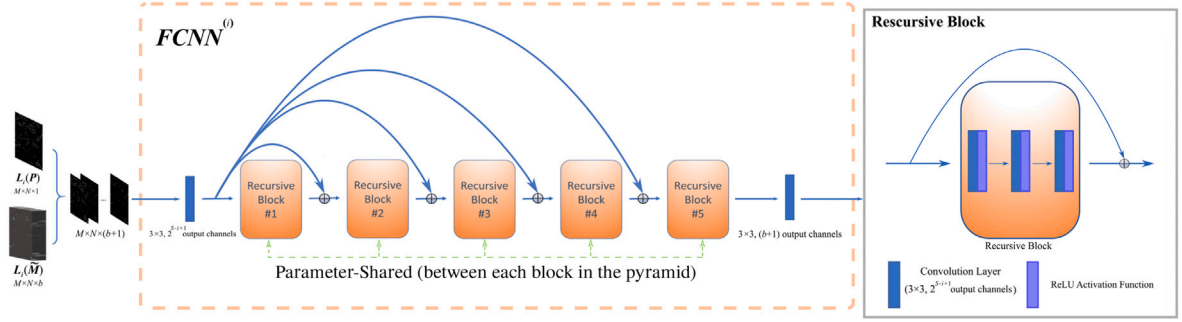

Fig. 5. The flowchart of the fusion convolution neural network (FCNN). In which, the output channels are set as 2^{S-i+1}, where S represents the total number of LPD layers, and the i is the corresponding LPD layer number.

4. Experimental results

In this section, we compare the proposed LPPN method with some recent state-of-the-art pansharpening approaches belonging to CS, MRA, VO, and DL categories. The employed sensors, the benchmarking methods, and the adopted quality indexes are described first. Afterwards, the experimental analysis both at reduced and full resolutions is reported.

4.1. Datasets

Several datasets have been acquired by the WorldView-3 sensor, which simultaneously captures a high resolution PAN channel and eight MS bands. Four standard colors (red, green, blue, and near-infrared 1) and four new bands (coastal, yellow, red edge, and near-infrared 2) are acquired. The images are distributed with a pixel size of 0.3 m and 1.2 m for PAN and MS, respectively. The spatial resolution ratio is equal to 4. The radiometric resolution is 11 bits.

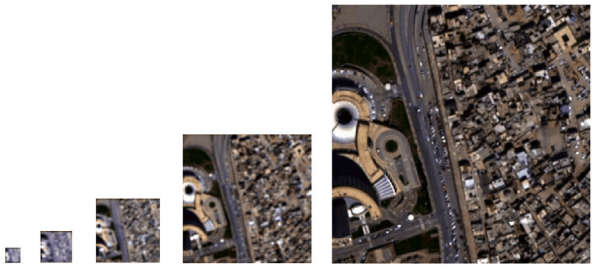

Fig. 6. The outputs of all the S layers of our LPPN, i.e., $\mathbf{Out}^{(i)}(\theta), i = 1, 2, \ldots, S$ (from right to left) with $S = 5$ on an exemplary real-world test case.

Different from WorldView-3, the images obtained by QuickBird and GaoFen-2 sensors consist of four MS bands and one PAN channel. The

Table 1
Details of experimental datasets.

Satellite Sensors		Spatial Dimension	Spectral Dimension	Radiometric Resolution	Themiatic Scenes	Original data Volume
WorldView-3	PAN	0.3 m	One band	11 bits	mixed (urban, vegetation, water scenario)	6
	MS	1.2 m	Eight bands			
QuickBird	PAN	0.6 m	One band	11 bits		1
	MS	2.4 m	Four bands			
GaoFen-2	PAN	1 m	One band	10 bits		1
	MS	4 m	Four bands			

Table 2
Quality assessment at reduced resolution on 1258 WorldView-3 test cases. The mean and the standard deviation indexes are used to sum up the obtained performance. Best results are in boldface.

Method	SAM	ERGAS	SCC	Q8	Qave	A.T.
EXP	5.85 ± 1.99	7.04 ± 2.93	0.660 ± 0.106	0.627 ± 0.142	0.640 ± 0.157	**0.016**
BDSD	6.90 ± 2.73	5.15 ± 2.27	0.878 ± 0.075	0.817 ± 0.118	0.823 ± 0.119	0.019
CNMF	5.53 ± 1.88	4.62 ± 1.93	0.888 ± 0.068	0.822 ± 0.123	0.825 ± 0.125	0.035
GLP_CBD	5.29 ± 1.96	4.16 ± 1.77	0.890 ± 0.070	0.854 ± 0.114	0.849 ± 0.123	0.033
CVPR19	5.21 ± 1.87	5.14 ± 2.12	0.867 ± 0.604	0.793 ± 0.123	0.788 ± 0.130	1.731
DiCNN	4.25 ± 1.35	3.05 ± 1.06	0.945 ± 0.047	0.893 ± 0.118	0.908 ± 0.115	0.838
PanNet	4.10 ± 1.30	2.96 ± 1.00	0.949 ± 0.046	0.896 ± 0.116	0.910 ± 0.116	0.863
DMDNet	3.97 ± 1.25	2.86 ± 0.97	0.953 ± 0.045	0.900 ± 0.114	0.912 ± 0.115	0.951
LPPN	**3.90** ± 1.29	**2.64** ± 0.96	**0.955** ± 0.045	**0.913** ± 0.111	**0.913** ± 0.114	0.977

spatial resolution ratio is equal to 4, again. In particular, QuickBird and GaoFen-2 data have a spatial resolution of about 0.6 m and 1 m for the PAN channel, respectively. Moreover, they have a radiometric resolution of 11 bits and 10 bits, respectively. Please refer to Table 1 for the summary of the experimental datasets.

4.2. Benchmark

The benchmark consists of one representative CS based method (*i.e.,* BDSD [10]), one representative MRA based methods (i.e., MTF_GLP_CBD [61], denoted as GLP_CBD from hereon for saving space in the tables), two regularization-based (VO) methods (i.e., CNMF [34] and CVPR19 [65]), and three state-of-the-art DL methods (i.e., PanNet [46], DiCNN [47], and DMDNet [11]).[3]

For fair comparison, all the compared DL-based methods (i.e., LPPN, DiCNN, PanNet, and DMDNet) are trained on the same training data using Python 3.7.4 with Tensorflow 1.14.0 on a desktop computer equipped with a Linux operating system and a GPU NVIDIA GeForce GTX 2080Ti with 11 GB.

4.3. Quality assessment indexes

For quantitative evaluation, we adopt the spectral angle mapper (SAM) [66] to evaluate the spectral quality, the erreur relative globale adimensionnelle de synthèse (ERGAS) index [67] as an extension of the root mean square error for multidimensional arrays, the spatial correlation coefficient (SCC) [68] in order to assess the spatial quality, and two universal image quality indexes [69], Qavg (an average version of the Q index along the spectral bands) and $Q2^n$ (Q4 and Q8 for four and eight bands datasets, respectively) representing the multidimensional extension of the Q index [70,71]. These indexes can be used when a reference (GT) image is available (i.e., at reduced resolution). Instead, when we need to assess the performance at full resolution, quality without reference indexes should be used [72–74]. In this paper, the quality with no reference (QNR) [72] index is exploited. It is obtained by the combination of the spatial distortion index, D_S, and the spectral distortion index, D_λ. The ideal values for SCC, QNR, Qave, and Q4/Q8 are 1. Instead, for ERGAS, SAM, D_λ, and D_S are 0. Furthermore, we exhibit the average running time for each fusion method, denoted as A.T. in seconds in the results tables.

4.4. Parameters tuning

In our LPPN network, we empirically employ the Adam [75] approach with a learning rate equal to 0.003 in order to minimize the loss function in (7). The number of iterations for the training step is 1×10^5 and the batch size is 32. Additionally, we set the kernel size of all filters as 3×3. In particular, for the setting of the other compared methods, we use the default asset pointing out in the related papers or source codes. In the DiCNN case, the batch size is set to 64 and the number of iterations is 3×10^5. Instead, for PanNet and DMDNet, the batch size is set to 32 and the number of iterations is 255,000. Under the use of the above-mentioned settings, the three methods can achieve their best performance.

4.5. Reduced resolution assessment

In this section, we assess the qualitative and quantitative performance of the compared methods on the reduced resolution datasets. The process for simulating the testing data is the same as that of the training data (see Section 3.3 for details). More in detail, the spatial size of the testing PAN/LRMS/GT patch is 64×64, 16×16, and 64 × 64 for WorldView-3 datasets in Table 2. For Figs. 7–9, the testing images for WorldView-3, QuickBird and GaoFen-2 cases are with the spatial size of 256×256, 64×64, and 256 × 256, respectively. We tested the compared approaches on a large dataset consisting of 1258 test cases extracted from WorldView-3 data, and Table 2 exhibits the average performance and the corresponding standard deviations for all the compared methods. From Table 2, it can be readily got that our method obtains high performance for all the quality indexes. We depict a typical result in Fig. 7. It can be seen that BDSD and CNMF show greater spectral distortion. GLP_CBD and CVPR19 have various blurring defects, especially visible in the building area at the lower right corner. Furthermore, the other three DL-based methods, i.e., PanNet, DMDNet and DiCNN, exhibit competitive visual performance. However, they still fail to outperform our method. To highlight the differences, we depicted several magnified sub-regions among the compared methods. It is clear

[3] The source codes of BDSD, GLP_CBD and CNMF can be downloaded at the website http://openremotesensing.net/kb/codes/pansharpening/. Additionally, the source codes of CVPR19 and PanNet can be downloaded at the website https://xueyangfu.github.io/. Instead, the source codes of DiCNN and DMDNet are not available online, thus we re-implemented them by ourselves using the default parameters indicated in the related papers to ensure their best performance.

Table 3
Quality assessment at reduced resolution on 25 QuickBird test cases. The mean and the standard deviation indexes are used to sum up the obtained performance. Best results are in boldface.

Method	SAM	ERGAS	SCC	Q4	Qave	A.T.
EXP	7.27 ± 2.31	10.88 ± 2.77	0.530 ± 0.023	0.545 ± 0.139	0.540 ± 0.146	**0.008**
BDSD	6.71 ± 2.08	7.05 ± 1.12	0.840 ± 0.084	0.769 ± 0.178	0.761 ± 0.187	0.205
CNMF	6.34 ± 2.75	7.07 ± 2.77	0.771 ± 0.295	0.683 ± 0.281	0.680 ± 0.280	0.172
GLP_CBD	6.52 ± 1.96	6.91 ± 1.06	0.840 ± 0.085	0.779 ± 0.169	0.764 ± 0.185	0.036
CVPR19	6.84 ± 2.14	8.62 ± 1.86	0.815 ± 0.060	0.686 ± 0.171	0.676 ± 0.179	4.478
DiCNN	4.79 ± 1.09	5.06 ± 0.57	0.908 ± 0.067	0.835 ± 0.187	0.830 ± 0.194	0.868
PanNet	4.78 ± 1.07	4.80 ± 0.34	0.915 ± 0.078	0.841 ± 0.184	0.841 ± 0.188	0.887
DMDNet	4.61 ± 0.94	4.46 ± 0.29	0.919 ± 0.087	0.845 ± 0.193	0.845 ± 0.197	1.047
LPPN	**4.39** ± 0.85	**4.41** ± 0.50	**0.937** ± 0.065	**0.851** ± 0.185	**0.848** ± 0.190	1.122

Table 4
Quality assessment at reduced resolution on 25 GaoFen-2 test cases. The mean and the standard deviation indexes are used to sum up the obtained performance. Best results are in boldface.

Method	SAM	ERGAS	SCC	Q4	Qave	A.T.
EXP	2.88 ± 0.47	3.58 ± 0.44	0.690 ± 0.047	0.760 ± 0.030	0.773 ± 0.032	**0.017**
BDSD	2.90 ± 0.43	2.53 ± 0.46	0.859 ± 0.054	0.873 ± 0.047	0.884 ± 0.042	0.019
CNMF	3.20 ± 0.56	2.74 ± 0.59	0.860 ± 0.055	0.852 ± 0.042	0.877 ± 0.044	0.455
GLP_CBD	2.83 ± 0.50	2.49 ± 0.44	0.852 ± 0.054	0.873 ± 0.042	0.877 ± 0.039	0.079
CVPR19	2.57 ± 0.44	2.76 ± 0.37	0.854 ± 0.041	0.861 ± 0.024	0.861 ± 0.024	9.037
DiCNN	1.77 ± 0.31	1.57 ± 0.20	0.943 ± 0.013	0.949 ± 0.015	0.953 ± 0.012	0.879
PanNet	1.65 ± 0.25	1.44 ± 0.12	0.955 ± 0.009	0.951 ± 0.022	0.963 ± 0.011	0.887
DMDNet	1.54 ± 0.24	1.32 ± 0.12	0.961 ± 0.009	0.956 ± 0.022	0.968 ± 0.012	1.049
LPPN	**1.49** ± 0.20	**1.18** ± 0.12	**0.968** ± 0.007	**0.968** ± 0.014	**0.970** ± 0.015	1.129

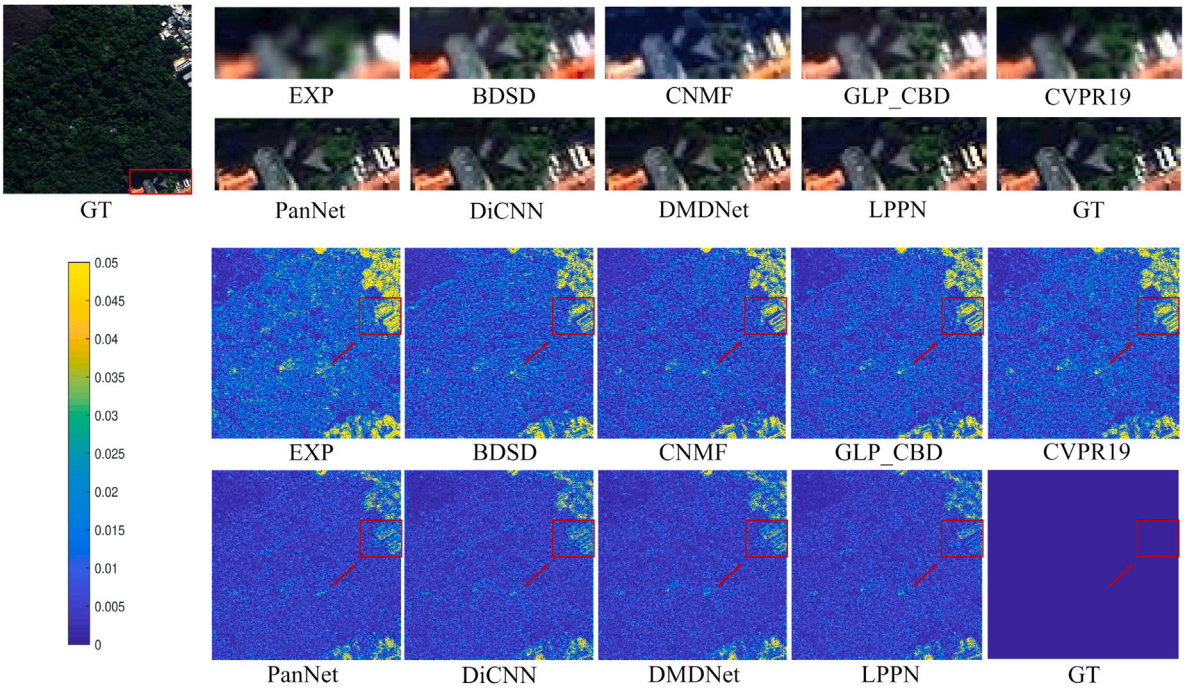

Fig. 7. The visual comparisons on a reduced resolution WorldView-3 case (depicted bands: 1, 3 and 5). First two rows: The fusion results by means of BDSD, CNMF, GLP_CBD, CVPR19, DiCNN, PanNet, DMDNet, and Proposed LPPN. Third and fourth rows: The corresponding residual maps using the GT image as reference. To aid the visual inspection, we display the residual maps obtained by the analysis of the third spectral band.

that our method obtains the best visual performance, closer to the GT image in spatial aspects, including sharper edges and clearer objects. This can be observed in the lower right area of the sub-regions.(e.g., see the orange roofs of the close-ups in Fig. 7). Moreover, we compute the residual maps between the comparison candidates and the ground-truth (GT) image. It is clear that the fusion image of the proposed approach is closer to GT image, getting a residual map close to zero almost everywhere. For average running time in all testing scenarios, EXP method achieves the fastest time but the worst performance among the compared methods. Concerning conventional methods, CVPR19 lasts the longest for its iteration-heavy dependency. Regarding to DL-based methods, DiCNN and PanNet get the best running times, followed by the DMDNet and the proposed approach.

In order to corroborate the results obtained on the WorldView-3 test cases, we assessed the performance of the compared approaches on data acquired by the QuickBird sensor (Indianapolis datasets) and GaoFen-2 sensor (Beijing and Guangzhou datasets[4]). For the best performance, we adjusted its output channels in the head of each FCNN to 2^{S-i+2} for both datasets. The deep learning-based methods are properly trained

[4] Datasets from: http://www.rscloudmart.com/dataProduct/sample.

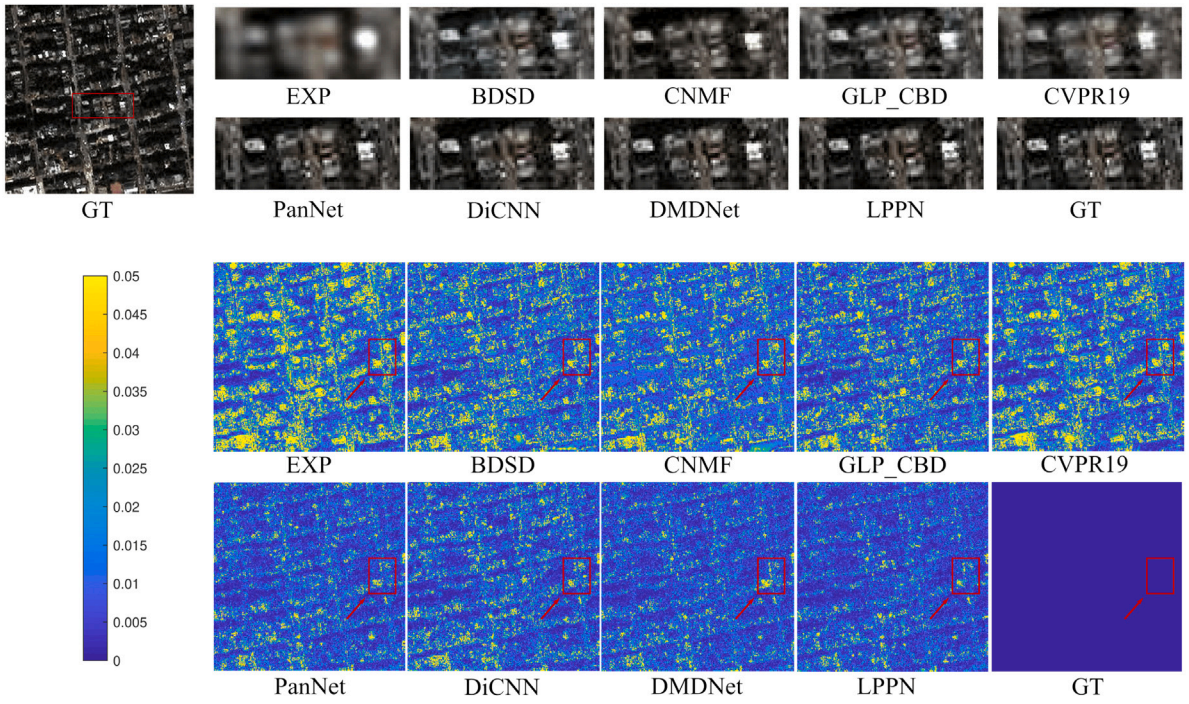

Fig. 8. The visual comparisons on a reduced resolution QuickBird case (depicted bands: 1, 2 and 3). First two rows: The fusion results by means of BDSD, CNMF, GLP_CBD, CVPR19, DiCNN, PanNet, DMDNet, and Proposed LPPN. Third and fourth rows: The corresponding residual maps using the GT image as reference. To aid the visual inspection, we display the residual maps obtained by the analysis of the second spectral band.

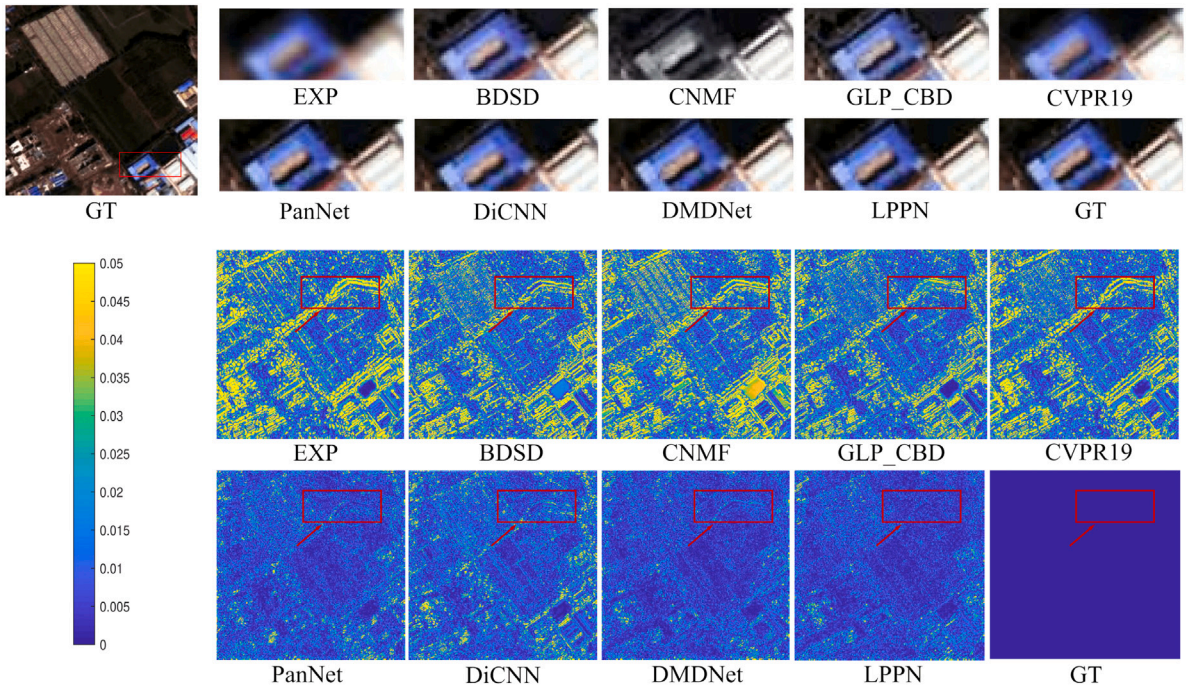

Fig. 9. The visual comparisons on a GaoFen-2 case (depicted bands: 1, 2 and 3). First two rows: The fusion results by means of BDSD, CNMF, GLP_CBD, CVPR19, DiCNN, PanNet, DMDNet, and Proposed LPPN. Third and fourth rows: The corresponding residual maps using the GT image as reference. To aid the visual inspection, we display the residual maps obtained by the analysis of the fourth spectral band.

on training data acquired by these sensors. Similar to WorldView-3 testing cases, we also compare the approaches on larger datasets of 256×256 spatial size consisting of 25 and 25 test cases extracted from QuickBird and GaoFen-2 datasets, respectively. Tables 3 and 4 exhibit the average performance and the corresponding standard deviations for all the compared methods, and Figs. 8 and 9 show the typical corresponding performance of the compared methods. In the scenario of the QuickBird sensor, BDSD and CNMF demonstrate acceptable visual outcomes. However, GLP_CBD and CVPR19 still exhibit blurring defects on the white roof-like object, thus decreasing the spatial performance. For DL-based methods, the outcomes are still competitive. With the aid of the residual maps, we can see our method outperforms the other DL-based compared methods for its close degree of proximity to the GT image. In the GaoFen-2 test case, BDSD and CNMF show acceptable

Table 5
Quality assessment at full resolution on 200 WorldView-3 test cases. The mean and the standard deviation indexes are used to sum up the obtained performance. Best results are in boldface.

Method	QNR	D_λ	D_S	A.T.
EXP	0.913 ± 0.031	**0.000** ± 0.000	0.086 ± 0.030	**0.018**
BDSD	0.893 ± 0.032	0.033 ± 0.013	0.077 ± 0.027	0.035
CNMF	0.896 ± 0.072	0.040 ± 0.037	0.067 ± 0.047	0.416
GLP_CBD	0.920 ± 0.050	0.028 ± 0.024	0.055 ± 0.032	0.079
CVPR19	0.932 ± 0.023	0.012 ± 0.006	0.057 ± 0.019	8.964
DiCNN	0.953 ± 0.036	0.018 ± 0.020	0.030 ± 0.020	0.861
PanNet	0.961 ± 0.021	0.019 ± 0.009	0.020 ± 0.012	0.886
DMDNet	0.960 ± 0.020	0.019 ± 0.010	0.021 ± 0.012	1.038
LPPN	**0.963** ± 0.023	0.018 ± 0.012	**1.089** ± 0.013	1.082

Table 6
Quality assessment at full resolution on 20 QuickBird test cases. The mean and the standard deviation indexes are used to sum up the obtained performance. Best results are in boldface.

Method	QNR	D_λ	D_S	A.T.
EXP	0.845 ± 0.026	**0.000** ± 0.000	0.156 ± 0.026	**0.008**
BDSD	0.877 ± 0.029	0.034 ± 0.016	0.092 ± 0.036	0.025
CNMF	0.779 ± 0.072	0.076 ± 0.050	0.067 ± 0.047	0.144
GLP_CBD	0.832 ± 0.028	0.055 ± 0.013	**0.012** ± 0.021	0.043
CVPR19	0.936 ± 0.013	0.008 ± 0.004	0.056 ± 0.012	4.539
DiCNN	0.910 ± 0.027	0.026 ± 0.009	0.065 ± 0.024	0.872
PanNet	0.943 ± 0.021	0.027 ± 0.008	0.031 ± 0.015	0.899
DMDNet	0.943 ± 0.020	0.024 ± 0.008	0.034 ± 0.014	1.051
LPPN	**0.947** ± 0.004	0.025 ± 0.009	0.029 ± 0.007	1.132

Table 7
Quality assessment at full resolution on 20 GaoFen test cases. The mean and the standard deviation indexes are used to sum up the obtained performance. Best results are in boldface.

Method	QNR	D_λ	D_S	A.T.
EXP	0.860 ± 0.269	**0.000** ± 0.000	0.140 ± 0.269	**0.008**
BDSD	0.882 ± 0.035	0.021 ± 0.014	0.099 ± 0.029	0.201
CNMF	0.779 ± 0.043	0.079 ± 0.030	0.067 ± 0.047	0.172
GLP_CBD	0.892 ± 0.037	0.048 ± 0.021	0.064 ± 0.021	0.034
CVPR19	0.832 ± 0.035	0.006 ± 0.005	0.163 ± 0.036	4.479
DiCNN	0.870 ± 0.027	0.033 ± 0.013	0.100 ± 0.022	0.873
PanNet	0.970 ± 0.007	0.012 ± 0.006	0.019 ± 0.008	0.881
DMDNet	0.969 ± 0.007	0.010 ± 0.006	0.021 ± 0.006	1.045
LPPN	**0.970** ± 0.004	0.010 ± 0.005	**0.018** ± 0.005	1.134

visual results, again. GLP_CBD and CVPR19 still suffer from spatial distortion. However, in this case, the CNMF method demonstrates an evident spectral distortion. For DL-based methods, we can observe that our method yields better outcomes with the aid of the residual maps. In both figures, the proposed LPPN clearly shows its spatial advantages thanks to lower image residuals (see the close-ups in the related figures), thus very high performance of the proposed LPPN method can be easily observed.

4.6. Full resolution assessment

We also compare our LPPN approach with recent state-of-the-art pansharpening approaches on full resolution WorldView-3, QuickBird and GaoFen-2 data, whose PAN/LRMS patch is of spatial size 256 × 256 and 64 × 64, respectively. Due to the lack of a reference (GT) image, the QNR index is used instead of the quality indexes in Section 4.5. Tables 5–7 report the outcomes respectively on 200 real WorldView-3, 20 real QuickBird and 20 GaoFen-2 examples, synthesizing them using the mean and the standard deviation operators. It is worth to be pointed out that the proposed method gets the best results related to the overall quality index at full resolution, QNR. The other two indexes, *i.e.,* D_λ and D_S, are also close to the approaches that get the best performance. Moreover, we also show the visual comparison on a full resolution WorldView-3 datasets in Fig. 10, in which the LPPN approach yields more image details and sharper image edges than the compared methods.

Table 8
Performance assessment by varying the layer number of the proposed Laplacian pyramid-based approach.

Layers	SAM	ERGAS	SCC	Q8	Qave
1	4.81	3.29	0.9584	0.9467	0.9480
2	4.54	3.18	0.9624	0.9496	0.9515
3	4.41	3.12	0.9627	0.9507	0.9520
4	4.40	3.11	0.9640	0.9509	0.9522
5	**4.30**	**3.10**	**0.9657**	**0.9518**	**0.9536**
6	4.52	3.16	0.9621	0.9506	0.9519

Table 9
Performance assessment by varying the FCNN structure.

Methods	SAM	ERGAS	SCC	Q8	Qave
Without recursive	4.83	3.54	0.9442	0.942	0.943
Fixed Gaussian kernel	4.66	3.21	0.9616	0.947	0.950
ℓ_1 loss	4.51	3.18	0.9617	0.948	0.949
Proposed	**4.30**	**3.10**	**0.9657**	**0.952**	**0.954**

4.7. Discussions

In this section, we will deeply discuss about the proposed architecture (i.e., the number of Laplacian layers, the effect of recursive blocks, the use of kernels, and loss functions) after comparing the convergence of different fusion methods. Note that for the discussion of this part, we take a WorldView-3 datasets (spatial size 256 × 256) as the test example for the sake of brevity.

4.7.1. Convergence analysis

We utilize the mean squared error (MSE) loss on WorldView-3's 8,806 training data to compare the different network convergence among all the compared DL-based methods, i.e., DiCNN, PanNet, DMDNet and our LPPN architecture. As shown in Fig. 11, our network has demonstrated a lower training error with less iterations.

4.7.2. The influence of the Laplacian pyramid layer number

A key parameter of Laplacian pyramids is the layer number. Since the maximum spatial size of our training images is 64 × 64, S can only vary from 1 to 6. Using the datasets depicted in Fig. 7, the results of our method by varying the layer number are reported in Table 8. It is easy to show that the best performance can be obtained by the 5 layers configuration, i.e. the one adopted in this paper.

4.7.3. FCNN structure discussion

After determining the network levels, we look at the sub-network level, that is the FCNN. We studied several configurations of the FCNN. In particular, we have:

(a) With recursive Vs. without recursive blocks: In order to investigate the effects of the recursive blocks, we substituted the recursive blocks with ResNet [62] blocks in each FCNN, as depicted in Fig. 12. Having a look at the first and fourth rows in Table 9, it is straightforward that the proposed architecture with recursive blocks generates better results than that of the one without recursive blocks. Moreover, the amount of parameters related to the compared architectures is 174086 (without recursive) against 50706 (with recursive), and the average usage of GPU RAM is 1485 MB (without recursive) against 1453 MB (with recursive), considering the test case depicted in Fig. 7, thus demonstrating the reduction of the number of parameters under the use of recursive blocks.

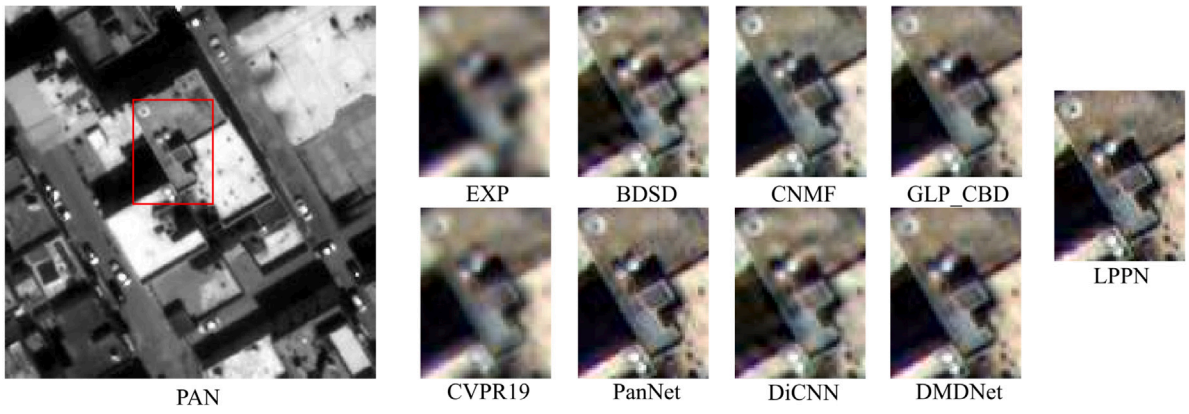

Fig. 10. The visual comparisons on a full resolution WorldView-3 case (depicted bands: 1, 3, and 5).

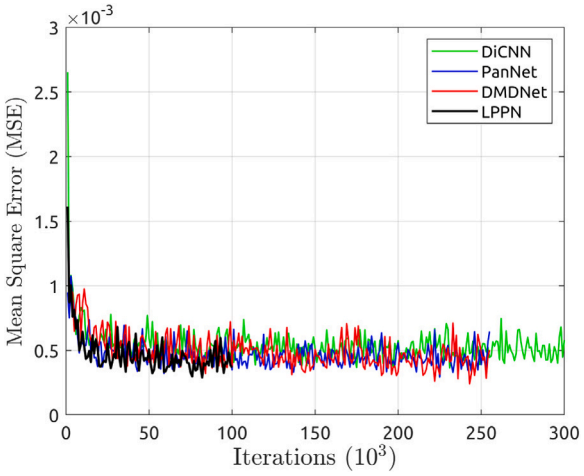

Fig. 11. Convergence of compared DL-based methods.

Table 10
Comparison of the number of parameters (NoPs) , total training time in seconds for all the CNN-based methods.

	DiCNN	PanNet	DMDNet	LPPN
NoPs	1.5×10^5	2.5×10^5	3.1×10^5	$\mathbf{0.5 \times 10^5}$
Time	$\mathbf{3.7 \times 10^3}$	4.8×10^3	1.2×10^4	7.4×10^3

Table 11
Generalization ability assessment on the WorldView-2 dataset.

Method	SAM	ERGAS	SCC	Q4	Qave
EXP	8.24	8.88	0.4958	0.6468	0.6667
BDSD	8.42	6.30	0.7989	0.8400	0.8479
CNMF	7.41	6.29	0.8453	0.8298	0.8365
GLP_CBD	7.77	6.31	0.8046	0.8370	0.8390
CVPR19	7.30	6.89	0.8144	0.7882	0.7976
DiCNN	6.80	5.54	0.8680	0.8493	0.8658
PanNet	6.52	5.40	0.8588	0.8643	0.8785
DMDNet	**6.37**	**5.22**	0.8648	**0.8707**	**0.8824**
LPPN	6.50	5.38	**0.8742**	0.8641	0.8725

(b) Fixed Gaussian Vs. MTF kernels: To verify our statement in Section 3.1.1, we change the shape of the filters by using classical fixed Gaussian kernel instead of MTF kernels to deal with the generation of the Laplacian pyramid. Having a look at the second and fourth rows Table 9, it is clear that the proposed architecture with MTF-based kernels used for convolution obtains the best quantitative performance.

(c) ℓ_1 Vs. ℓ_2 loss functions: We also consider the performance of our network architecture with ℓ_1 and ℓ_2 loss functions. The results shown in the third and fourth rows demonstrate that the proposed ℓ_2 loss function is advisable.

4.7.4. Generalization ability assessment

We also evaluate the generation ability of our network architecture among the aforementioned fusion methods by utilizing WorldView-2 datasets, whose number of spectral bands are equal to the WorldView-3 ones. The datasets can be freely downloaded from the same link as for the WorldView-3 datasets. More in detail, the testing WorldView-2 datasets have the PAN image of spatial size of 256 × 256 with a 4 PAN/MS spatial ratio. The outcomes are reported in Table 11. From the table, we can see that albeit the performance of our method does not outstrip the DMDNet, they are very close to the best. For the reason, we believe that the high-pass filters employed by PanNet and DMDNet efficiently extract more high-frequency features, which are crucial to restore the original fused information.

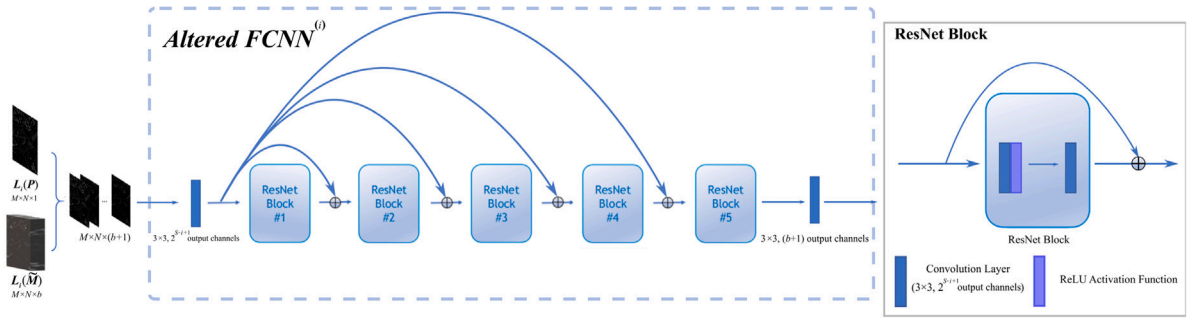

Fig. 12. Altered ResNet block structure of the FCNN.

4.7.5. Number of parameters and total training time assessment

Lastly, we compare the number of parameters and total training time of the four DL-based algorithms. As a consequence of the use of the recursive structure in each FCNN, our complexity significantly outperforms the one of the other methods. Table 10 shows the comparison of the number of parameters (NoPs) for all the CNN methods on the WorldView-3 datasets. From this table, it is clear that the proposed LPPN method has only 0.5×10^5 parameters, which is significantly less than the NoPs of DiCNN (1.8×10^5), PanNet (2.5×10^5), and DMDNet (3.2×10^5). To the best of our knowledge, the proposed LPPN is one of the best lightweight network for pansharpening showing a small NoPs and also getting state-of-the-art performance. As for total training time, DiCNN method yields the shortest one. We believed the reason for this is the relatively simple network structure, which only contains 3 convolutional blocks and a skip-connection composition.

5. Conclusions

In this paper, we proposed an efficient deep pyramid network architecture for pansharpening. The network architecture consists of three parts. The first one is the decomposition of the input image set into Laplacian pyramids using MTF-based kernels. Afterwards, these pyramids are fused by the fusion convolutional neural network. Finally, we reconstruct the multi-layer outputs comparing them with the reference (GT) data exploiting an ℓ_2 loss function in order to train the network. A broad experimental analysis demonstrates that the proposed approach outperforms the compared state-of-the-art pansharpening methods. Furthermore, some discussions about the network convergence, the number of Laplacian pyramid layers, the influence of the loss function, the use of recursive blocks, the generalization ability and so forth, are provided to the readers. Finally, an analysis on the number of parameters and total training time of the network pointed out that our LPPN is a lightweight pansharpening network getting state-of-the-art performance.

CRediT authorship contribution statement

Cheng Jin: Methodology, Software, Writing – original draft. **Liang-Jian Deng:** Methodology, Supervision, Writing – review & editing. **Ting-Zhu Huang:** Supervision, Funding acquisition. **Gemine Vivone:** Writing – review & editing.

Declaration of competing interest

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

Acknowledgments

The work is supported by National Natural Science Foundation of China grants 61702083 and 12001446, Key Projects of Applied Basic Research in Sichuan Province (Grant No. 2020YJ0216), and National Key Research and Development Program of China (Grant No. 2020YFA0714001).

References

[1] V. Søren, M. Joav, Meta-analysis of positive effects, side effects and adverse events of holistic mind-body medicine (clinical holistic medicine): Experience from Denmark, Sweden, United Kingdom and Germany, Int. J. Adolesc. Med. Health 21 (4) (2009) 441–456.

[2] Z. Yong, Recent developments in technology and language learning: A literature review and meta-analysis, CALICO J. (2003) 7–27.

[3] W. Xu, W. Yuan, W. Dong, J. Xia, D. Liu, Y. Chen, A meta-analysis of the response of soil moisture to experimental warming, Environ. Res. Lett. 8 (4) (2013) 044027.

[4] D. Mauro, P. Saurabh, P. Fabio, G. Paulo, C. Jocelyn, B. Atli, Challenges and opportunities of multimodality and data fusion in remote sensing, Proc. IEEE 103 (9) (2015) 1585–1601.

[5] T. Claire, R. Thierry, W. Lucien, C. Jocelyn, Synthesis of multispectral images to high spatial resolution: A critical review of fusion methods based on remote sensing physics, IEEE Trans. Geosci. Remote Sens. 46 (5) (2008) 1301–1312.

[6] G. Vivone, L. Alparone, J. Chanussot, M.D. Mura, A. Garzelli, G.A. Licciardi, R. Restaino, L. Wald, A critical comparison among pansharpening algorithms, IEEE Trans. Geosci. Remote Sens. 53 (5) (2014) 2565–2586.

[7] G. Vivone, M. Mura, A. Garzelli, R. Restaino, G. Scarpa, M. Ulfarsson, L. Alparone, J. Chanussot, A new benchmark based on recent advances in multispectral pansharpening: Revisiting pansharpening with classical and emerging pansharpening methods, IEEE Geosci. Remote Sens. Mag. (2020) http://dx.doi.org/10.1109/MGRS.2020.3019315.

[8] D. Di, W. Di, Comparisons of ERDAS and ENVI in thematic mapping, in: IEEE International Conference on Communication Software and Networks, ICCSN, 2011, pp. 517–520.

[9] S. Carlos, F. Laurel, S. Moreira, R. Dar, Mapping forest degradation in the eastern amazon from SPOT 4 through spectral mixture models, Remote Sens. Environ. 87 (4) (2003) 494–506.

[10] A. Garzelli, F. Nencini, L. Capobianco, Optimal MMSE pan sharpening of very high resolution multispectral images, IEEE Trans. Geosci. Remote Sens. 46 (1) (2008) 228–236.

[11] X. Fu, W. Wang, Y. Huang, X. Ding, J. Paisley, Deep multi-scale detail networks for multi-band spectral image sharpening, IEEE Trans. Neural Netw. Learn. Syst. (2020) http://dx.doi.org/10.1109/TNNLS.2020.2996498.

[12] J. Choi, K. Yu, Y. Kim, A new adaptive component-substitution based satellite image fusion by using partial replacement, IEEE Trans. Geosci. Remote Sens. 49 (2011) 295–309.

[13] A. Laben, V. Brower, Process for enhancing the spatial resolution of multispectral imagery using pan-sharpening, US Patent 6011875, 2000.

[14] X. Otazu, M. González-Audicana, O. Fors, J. Nunez, Introduction of sensor spectral response into image fusion methods. Application to wavelet-based methods, IEEE Trans. Geosci. Remote Sens. 43 (2005) 2376–2385.

[15] Z. Zhang, T.-Z. Huang, L.-J. Deng, Pansharpening via RoG-based filtering, in: IEEE International Geoscience and Remote Sensing Symposium, IGARSS, 2019.

[16] G. Vivone, Robust band-dependent spatial-detail approaches for panchromatic sharpening, IEEE Trans. Geosci. Remote Sens. 57 (9) (2019) 6421–6433.

[17] J. Liu, Smoothing filter-based intensity modulation: A spectral preserve image fusion technique for improving spatial details, Int. J. Remote Sens. 21 (18) (2000) 3461–3472.

[18] B. Aiazzi, L. Alparone, S. Baronti, A. Garzelli, Context-driven fusion of high spatial and spectral resolution images based on oversampled multiresolution analysis, IEEE Trans. Geosci. Remote Sens. 40 (2002) 2300–2312.

[19] R. Restaino, G. Vivone, P. Addesso, J. Chanussot, A pansharpening approach based on multiple linear regression estimation of injection coefficients, IEEE Geosci. Remote Sens. Lett. 17 (1) (2020) 102–106.

[20] G. Vivone, S. Marano, J. Chanussot, Pansharpening: Context-based generalized Laplacian pyramids by robust regression, IEEE Trans. Geosci. Remote Sens. 58 (9) (2020) 6152–6167.

[21] G. Vivone, R. Restaino, J. Chanussot, Full scale regression-based injection coefficients for panchromatic aharpening, IEEE Trans. Image Process. 27 (7) (2018) 3418–3431.

[22] G. Vivone, M. Simões, M. Dalla Mura, R. Restaino, J.M. Bioucas-Dias, G.A. Licciardi, J. Chanussot, Pansharpening based on semiblind deconvolution, IEEE Trans. Geosci. Remote Sens. 53 (4) (2015) 1997–2010.

[23] G. Vivone, P. Addesso, R. Restaino, M. Dalla Mura, J. Chanussot, Pansharpening based on deconvolution for multiband filter estimation, IEEE Trans. Geosci. Remote Sens. 57 (1) (2019) 540–553.

[24] G. Vivone, J. Chanussot, Fusion of short-wave infrared and visible near-infrared WorldView-3 data, Inf. Fusion 61 (2020) 71–83.

[25] G. Vivone, L. Alparone, A. Garzelli, S. Lolli, Fast reproducible pansharpening based on instrument and acquisition modeling: AWLP revisited, Remote Sens. 11 (19) (2019) 2315:1–2315:23.

[26] J. Saeedi, K. Faez, A new pan-sharpening method using multiobjective particle swarm optimization and the shiftable contourlet transform, ISPRS J. Photogramm. Remote Sens. 66 (3) (2011) 365–381.

[27] J. Liu, L.-J. Deng, F. Fang, T. Zeng, A rudin-osher-fatemi model-based pansharpening approach using RKHS and AHF representation, East Asian J. Appl. Math. 9 (2019) 13–27.

[28] L.-J. Deng, M. Feng, X.-C. Tai, The fusion of panchromatic and multispectral remote sensing images via tensor-based sparse modeling and hyper-Laplacian prior, Inf. Fusion 52 (2019) 76–89.

[29] V. Rosaria, R. Rocco, V. Gemine, D. Mauro, C. Jocelyn, A pansharpening method based on the sparse representation of injected details, IEEE Geosci. Remote Sens. Lett. 12 (1) (2014) 180–184.

[30] R. Gogineni, A. Chaturvedi, Sparsity inspired pan-sharpening technique using multi-scale learned dictionary, ISPRS J. Photogramm. Remote Sens. 146 (2018) 360–372.

[31] X.-Y. Fu, Z.-H. Lin, Y. Huang, X.-H. Ding, A variational pan-sharpening with local gradient constraints, in: 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR, 2019, pp. 10257–10266, http://dx.doi.org/10.1109/CVPR.2019.01051.

[32] X. Tian, Y.-R. Chen, C.-C. Yang, J.-Y. Ma, Variational pansharpening by exploiting cartoon-texture similarities, IEEE Trans. Geosci. Remote Sens. (2021) 1–16, http://dx.doi.org/10.1109/TGRS.2020.3048257.

[33] B. Coloma, C. Vicent, I. Laura, V. Joan, R. Bernard, A variational model for P+XS image fusion, Int. J. Comput. Vis. 69 (1) (2006) 43–58.

[34] Y. Naoto, Y. Takehisa, I. Akira, Coupled nonnegative matrix factorization unmixing for hyperspectral and multispectral data fusion, IEEE Trans. Geosci. Remote Sens. 50 (2) (2011) 528–537.

[35] L.-J. Deng, V. Gemine, W. Guo, M.D. Mura, J. Chanussot, A variational pan-sharpening approach based on reproducible kernel Hilbert apace and heaviside function, IEEE Trans. Image Process. 27 (2018) 4330–4344.

[36] Z.-C. Wu, T.-Z. Huang, L.-J. Deng, G. Vivone, J.-F.H. J. Miao, X.-L. Zhao, A new variational approach based on proximal deep injection and gradient intensity similarity for spatio-spectral image fusion, IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens. 13 (2020) 6277–6290.

[37] N. Yokoya, T. Yairi, A. Iwasaki, Coupled nonnegative matrix factorization unmixing for hyperspectral and multispectral data fusion, IEEE Trans. Geosci. Remote Sens. 50 (2) (2012) 528–537.

[38] R. Dian, S. Li, L. Fang, T. Lu, J.M. Bioucas-Dias, Nonlocal sparse tensor factorization for semiblind hyperspectral and multispectral image fusion, IEEE Trans. Cybern. 50 (10) (2020) 4469–4480.

[39] X. Meng, H. Shen, Q. Yuan, H. Li, L. Zhang, W. Sun, Pansharpening for cloud-contaminated very high-resolution remote sensing images, IEEE Trans. Geosci. Remote Sens. 57 (5) (2019) 2840–2854.

[40] G. Scarpa, S. Vitale, D. Cozzolino, Target-adaptive CNN-based pansharpening, IEEE Trans. Geosci. Remote Sens. 56 (9) (2018) 5443–5457.

[41] C. Liu, Y. Zhang, S. Wang, M. Sun, Y. Ou, Y. Wan, X. Liu, Band-independent encoder–decoder network for pan-sharpening of remote sensing images, IEEE Trans. Geosci. Remote Sens. 58 (7) (2020) 5208–5223.

[42] J. Peng, L. Liu, J. Wang, E. Zhang, X. Zhu, Y. Zhang, J. Feng, L. Jiao, PSMD-net: A novel pan-sharpening method based on a multiscale dense network, IEEE Trans. Geosci. Remote Sens. (2020) http://dx.doi.org/10.1109/TGRS.2020.3020162.

[43] H. Xu, J.-Y. Ma, Z.-F. Shao, H. Zhang, J.-J. Jiang, X.-J. Guo, SDPNet: A deep network for pan-sharpening with enhanced information representation, IEEE Trans. Geosci. Remote Sens. (2020) 1–15, http://dx.doi.org/10.1109/TGRS.2020.3022482.

[44] H. Zhang, J.-Y. Ma, GTP-pnet: A residual learning network based on gradient transformation prior for pansharpening, ISPRS J. Photogramm. Remote Sens. (ISSN: 0924-2716) 172 (2021) 223–239.

[45] G. Masi, D. Cozzolino, L. Verdoliva, G. Scarpa, Pansharpening by convolutional neural networks, Remote Sens. 8 (7) (2016) 594.

[46] J. Yang, X. Fu, Y. Hu, Y. Huang, X. Ding, J. Paisley, PanNet: A deep network architecture for pan-sharpening, in: IEEE International Conference on Computer Vision, ICCV, 2017.

[47] L. He, Y. Rao, J. Li, C. Jocelyn, P. Antonio, J. Zhu, B. Li, Pansharpening via detail injection based convolutional neural networks, IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens. 12 (4) (2019) 1188–1204.

[48] L.-J. Deng, G. Vivone, C. Jin, J. Chanussot, Detail injection-based deep convolutional neural networks for pansharpening, IEEE Trans. Geosci. Remote Sens. (2020) http://dx.doi.org/10.1109/TGRS.2020.3031366.

[49] J.-Y. Ma, W. Yu, C. Chen, P.-W. Liang, X.-J. Guo, J.-J. Jiang, Pan-GAN: An unsupervised pan-sharpening method for remote sensing image fusion, Inf. Fusion 62 (2020) 110–120.

[50] F. Ozcelik, U. Alganci, E. Sertel, G. Unal, Rethinking CNN-based pansharpening: Guided colorization of panchromatic images via GANs, IEEE Trans. Geosci. Remote Sens. (2020).

[51] X. Meng, Y. Xiong, F. Shao, H. Shen, W. Sun, G. Yang, Q. Yuan, R. Fu, H. Zhang, A large-scale benchmark data set for evaluating pansharpening performance: Overview and implementation, IEEE Geosci. Remote Sens. Mag. 9 (1) (2021) 18–52.

[52] H.-F. Shen, M.-H. Jiang, J. Li, Q.-Q. Yuan, Y.-C. Wei, L.-P. Zhang, Spatial–spectral fusion by combining deep learning and variational model, IEEE Trans. Geosci. Remote Sens. 57 (8) (2019) 6169–6181, http://dx.doi.org/10.1109/TGRS.2019.2904659.

[53] M.-H. Jiang, H.-F. Shen, J. Li, Q.-Q. Yuan, L.-P. Zhang, A differential information residual convolutional neural network for pansharpening, ISPRS J. Photogramm. Remote Sens. (ISSN: 0924-2716) 163 (2020) 257–271, http://dx.doi.org/10.1016/j.isprsjprs.2020.03.006.

[54] H. Zhang, H. Xu, Y. Xiao, X.-J. Guo, J.-Y. Ma, Rethinking the image fusion: A fast unified image fusion network based on proportional maintenance of gradient and intensity, in: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 34, No. 07, 2020, pp. 12797–12804, http://dx.doi.org/10.1609/aaai.v34i07.6975.

[55] W.-S. Lai, J.-B. Huang, A. Narendra, M.-H. Yang, Fast and accurate image super-resolution with deep Laplacian pyramid networks, IEEE Trans. Pattern Anal. Mach. Intell. 41 (11) (2019) 2599–2613.

[56] Y. Zhang, C. Liu, M. Sun, Y. Ou, Pan-sharpening using an efficient bidirectional pyramid network, IEEE Trans. Geosci. Remote Sens. 57 (8) (2019) 5549–5563.

[57] Q. Yuan, Y. Wei, X. Meng, H. Shen, L. Zhang, A multiscale and multidepth convolutional neural network for remote sensing imagery pan-sharpening, IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens. 11 (2018) 978–989.

[58] D. Jeong, Y. Kim, Deep learning based pansharpening using a Laplacian pyramid, in: Asian Conference on Remote Sensing, ACRS, 2019.

[59] O. Hannu, P. Peitsa, Gaussian pyramid wavelet transform for multiresolution analysis of images, Graph. Models Image Process. 58 (4) (1996) 394–398.

[60] B. Peter, A. Edward, The Laplacian pyramid as a compact image code, IEEE Trans. Commun. 31 (4) (1983) 532–540.

[61] B. Aiazzi, L. Alparone, S. Baronti, A. Garzelli, M. Selva, MTF-tailored multiscale fusion of high-resolution MS and pan imagery, Photogramm. Eng. Remote Sens. 72 (5) (2006) 591–596.

[62] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: IEEE Conference on Computer Vision and Pattern Recognition, CVPR, 2016, pp. 770–778.

[63] Y. Tai, J. Yang, X. Liu, Image super-resolution via deep recursive residual network, in: IEEE Conference on Computer Vision and Pattern Recognition, CVPR, 2017, pp. 3147–3155.

[64] L. Wald, T. Ranchin, M. Mangolini, Fusion of satellite images of different spatial resolutions: Assessing the quality of resulting images, Photogramm. Eng. Remote Sens. 63 (1997) 691–699.

[65] X. Fu, Z. Lin, Y. Huang, X. Ding, A variational pan-sharpening with local gradient constraints, in: IEEE Conference on Computer Vision and Pattern Recognition, CVPR, 2019, pp. 10265–10274.

[66] R. Yuhas, J. Boardman, A. Goetz, Determination of semi-arid landscape endmembers and seasonal trends using convex geometry spectral unmixing techniques, in: Summaries of the 4th Annual JPL Airborne Geoscience Workshop, 1993.

[67] W. Lucien, Data fusion: Definitions and architectures: Fusion of images of different spatial resolutions, 2002, Presses Des MINES.

[68] J. Zhou, D. Civco, J. Silander, A wavelet transform method to merge landsat TM and SPOT panchromatic data, Int. J. Remote Sens. 19 (1998) 743–757.

[69] W. Zhou, A. Bovik, A universal image quality index, IEEE Signal Process. Lett. 9 (3) (2002) 81–84.

[70] L. Alparone, S. Baronti, A. Garzelli, F. Nencini, A global quality measurement of pan-sharpened multispectral imagery, IEEE Geosci. Remote Sens. Lett. 1 (2004) 313–317.

[71] G. Andrea, N. Filippo, Hypercomplex quality assessment of multi/hyperspectral images, IEEE Geosci. Remote Sens. Lett. 6 (4) (2009) 662–665.

[72] A. Luciano, A. Bruno, B. Stefano, G. Andrea, N. Filippo, S. Massimo, Multispectral and panchromatic data fusion assessment without reference, Photogramm. Eng. Remote Sens. 74 (2) (2008) 193–200.

[73] M. Khan, L. Alparone, J. Chanussot, Pansharpening quality assessment using the modulation transfer functions of instruments, IEEE Trans. Geosci. Remote Sens. 11 (47) (2009) 3880–3891.

[74] G. Vivone, R. Restaino, J. Chanussot, A Bayesian procedure for full-resolution quality assessment of pansharpened products, IEEE Trans. Geosci. Remote Sens. 56 (8) (2018) 4820–4834.

[75] D. Kingma, B. Jimmy, Adam: A method for stochastic optimization, 2014, arXiv preprint: 1412.6980.