Multispectral and Hyperspectral Image Fusion by MS/HS Fusion Net

Qi Xie[1], Minghao Zhou[1], Qian Zhao[1], Deyu Meng[1,*], Wangmeng Zuo[2], Zongben Xu[1]
[1]Xi'an Jiaotong University; [2]Harbin Institute of Technology
xq.liwu@stu.xjtu.edu.cn woshizhouminghao@stu.xjtu.edu.cn timmy.zhaoqian@gmail.com
dymeng@mail.xjtu.edu.cn wmzuo@hit.edu.cn zbxu@mail.xjtu.edu.cn

Abstract

Hyperspectral imaging can help better understand the characteristics of different materials, compared with traditional image systems. However, only high-resolution multispectral (HrMS) and low-resolution hyperspectral (LrHS) images can generally be captured at video rate in practice. In this paper, we propose a model-based deep learning approach for merging an HrMS and LrHS images to generate a high-resolution hyperspectral (HrHS) image. In specific, we construct a novel MS/HS fusion model which takes the observation models of low-resolution images and the low-rankness knowledge along the spectral mode of HrHS image into consideration. Then we design an iterative algorithm to solve the model by exploiting the proximal gradient method. And then, by unfolding the designed algorithm, we construct a deep network, called MS/HS Fusion Net, with learning the proximal operators and model parameters by convolutional neural networks. Experimental results on simulated and real data substantiate the superiority of our method both visually and quantitatively as compared with state-of-the-art methods along this line of research.

1. Introduction

A hyperspectral (HS) image consists of various bands of images of a real scene captured by sensors under different spectrums, which can facilitate a fine delivery of more faithful knowledge under real scenes, as compared to traditional images with only one or a few bands. The rich spectra of HS images tend to significantly benefit the characterization of the imaged scene and greatly enhance performance in different computer vision tasks, including object recognition, classification, tracking and segmentation [10, 37, 35, 36].

In real cases, however, due to the limited amount of incident energy, there are critical tradeoffs between spatial and spectral resolution. Specifically, an optical system usually can only provide data with either high spatial resolution but a small number of spectral bands (e.g., the standard RGB image) or with a large number of spectral bands but reduced spatial resolution [23]. Therefore, the research issue on merging a high-resolution multispectral (HrMS) image and a low-resolution hyperspectral (LrHS) image to generate a high-resolution hyperspectral (HrHS) image, known as MS/HS fusion, has attracted great attention [47].

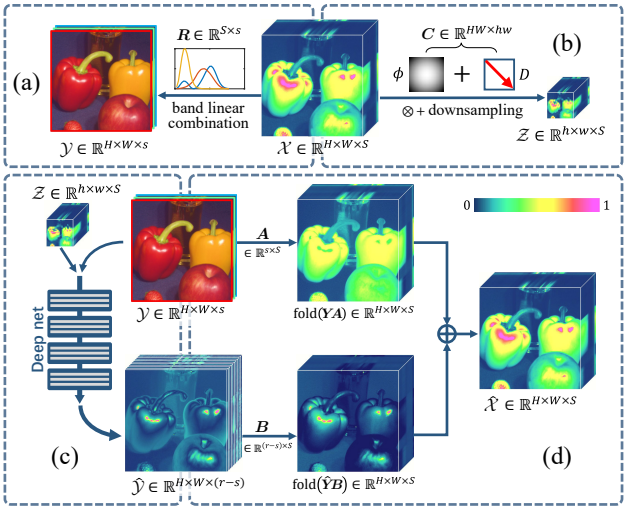


Figure 1. (a)(b) The observation models for HrMS and LrHS images, respectively. (c) Learning bases $\hat{\boldsymbol{Y}}$ by deep network, with HrMS $\boldsymbol{Y}$ and LrHS $\boldsymbol{Z}$ as the input of the network. (d) The HrHSI $\boldsymbol{X}$ can be linearly represented by $\boldsymbol{Y}$ and to-be-estimated $\hat{\boldsymbol{Y}}$, in a formulation of $\boldsymbol{X} \approx \boldsymbol{Y}\boldsymbol{A} + \hat{\boldsymbol{Y}}\boldsymbol{B}$, where the rank of $\boldsymbol{X}$ is r.

The observation models for the HrMS and LrHS images are often written as follows [12, 24, 25]:

$$\boldsymbol{Y} = \boldsymbol{X}\boldsymbol{R} + \boldsymbol{N}_y, \tag{1}$$
$$\boldsymbol{Z} = \boldsymbol{C}\boldsymbol{X} + \boldsymbol{N}_z, \tag{2}$$

where $\boldsymbol{X} \in \mathbb{R}^{HW\times S}$ is the target HrHS image[1] with H, W and S as its height, width and band number, respectively, $\boldsymbol{Y} \in \mathbb{R}^{HW\times s}$ is the HrMS image with s as its band number ($s < S$), $\boldsymbol{Z} \in \mathbb{R}^{hw\times S}$ is the LrHS image with h, w and S as its height, width and band number ($h < H$, $w < W$), $\boldsymbol{R} \in \mathbb{R}^{S\times s}$ is the spectral response of the multispectral sensor as shown in Fig. 1 (a), $\boldsymbol{C} \in \mathbb{R}^{hw\times HW}$ is a linear operator which is often assumed to be composed of a cyclic convolution operator ϕ and a down-sampling matrix $\boldsymbol{D}$ as shown in Fig. 1 (b), $\boldsymbol{N}_y$ and $\boldsymbol{N}_z$ are the noises contained in

*Corresponding author.

[1]The target HS image can also be written as tensor $\mathcal{X} \in \mathbb{R}^{H\times W\times S}$. We also denote the folding operator for matrix to tensor as: $\text{fold}(\boldsymbol{X}) = \mathcal{X}$.

HrMS and LrHS images, respectively. Many methods have been designed based on (1) and (2), and achieved good performance [40, 14, 24, 25].

Since directly recovering the HrHS image $\boldsymbol{X}$ is an ill-posed inverse problem, many techniques have been exploited to recover $\boldsymbol{X}$ by assuming certain priors on it. For example, [54, 2, 11] utilize the prior knowledge of HrHS that its spatial information could be sparsely represented under a dictionary trained from HrMS. Besides, [27] assumes the local spatial smoothness prior on the HrHS image and uses total variation regularization to encode it in their optimization model. Instead of exploring spatial prior knowledge from HrHS, [52] and [26] assume more intrinsic spectral correlation prior on HrHS, and use low-rank techniques to encode such prior along the spectrum to reduce spectral distortions. Albeit effective for some applications, the rationality of these techniques relies on the subjective prior assumptions imposed on the unknown HrHS to be recovered. An HrHS image collected from real scenes, however, could possess highly diverse configurations both along space and across spectrum. Such conventional learning regimes thus could not always flexibly adapt different HS image structures and still have room for performance improvement.

Methods based on Deep Learning (DL) have outperformed traditional approaches in many computer vision tasks [34] in the past decade, and have been introduced to HS/MS fusion problem very recently [28, 30]. As compared with conventional methods, these DL based ones are superior in that they need fewer assumptions on the prior knowledge of the to-be-recovered HrHS, while can be directly trained on a set of paired training data simulating the network inputs (LrHS&HrMS images) and outputs (HrHS images). The most commonly employed network structures include CNN [7], 3D CNN [28], and residual net [30]. Like other image restoration tasks where DL is successfully applied to, these DL-based methods have also achieved good resolution performance for MS/MS fusion task.

However, the current DL-based MS/HS fusion methods still have evident drawbacks. The most critical one is that these methods use general frameworks for other tasks, which are not specifically designed for MS/HS fusion. This makes them lack interpretability specific to the problem. In particular, they totally neglect the observation models (1) and (2) [28, 30], especially the operators $\boldsymbol{R}$ and $\boldsymbol{C}$, which facilitate an understanding of how LrHS and HrMs are generated from the HrHS. Such understanding, however, should be useful for calculating HrHS images. Besides this generalization issue, current DL methods also neglect the general prior structures of HS images, such as spectral low-rankness. Such priors are intrinsically possessed by all meaningful HS images, which implies that DL-based methods still have room for further enhancement.

In this paper, we propose a novel deep learning-based method that integrates the observation models and image prior learning into a single network architecture. This work mainly contains the following three-fold contributions:

Firstly, we propose a novel MS/HS fusion model, which not only takes the observation models (1) and (2) into consideration but also exploits the approximate low-rankness prior structure along the spectral mode of the HrHS image to reduce spectral distortions [52, 26]. Specifically, we prove that if and only if observation model (1) can be satisfied, the matrix of HrHS image $\boldsymbol{X}$ can be linearly represented by the columns in HrMS matrix $\boldsymbol{Y}$ and a to-be-estimated matrix $\hat{\boldsymbol{Y}}$, i.e., $\boldsymbol{X} = \boldsymbol{Y}\boldsymbol{A} + \hat{\boldsymbol{Y}}\boldsymbol{B}$ with coefficient matrices $\boldsymbol{A}$ and $\boldsymbol{B}$. One can see Fig. 1 (d) for easy understanding. We then construct a concise model by combining the observation model (2) and the linear representation of $\boldsymbol{X}$. We also exploit the proximal gradient method [3] to design an iterative algorithm to solve the proposed model.

Secondly, we unfold this iterative algorithm into a deep network architecture, called MS/HS Fusion Net or MHF-net, to implicitly learn the to-be-estimated $\hat{\boldsymbol{Y}}$, as shown in Fig. 1 (c). After obtaining $\hat{\boldsymbol{Y}}$, we can then easily achieve $\boldsymbol{X}$ with $\boldsymbol{Y}$ and $\hat{\boldsymbol{Y}}$. To the best of our knowledge, this is the first deep-learning-based MS/HS fusion method that fully considers the intrinsic mechanism of the MS/HS fusion problem. Moreover, all the parameters involved in the model can be automatically learned from training data in an end-to-end manner. This means that the spatial and spectral responses ($\boldsymbol{R}$ and $\boldsymbol{C}$) no longer need to be estimated beforehand as most of the traditional non-DL methods did, nor to be fully neglected as current DL methods did.

Thirdly, we have collected or realized current state-of-the-art algorithms for the investigated MS/HS fusion task, and compared their performance on a series of synthetic and real problems. The experimental results comprehensively substantiate the superiority of the proposed method, both quantitatively and visually.

In this paper, we denote scalar, vector, matrix and tensor in non-bold case, bold lower case, bold upper case and calligraphic upper case letters, respectively.

2. Related work

2.1. Traditional methods

The pansharpening technique in remote sensing is closely related to the investigated MS/HS problem. This task aims to obtain a high spatial resolution MS image by the fusion of a MS image and a wide-band panchromatic image. A heuristic approach to perform MS/HS fusion is to treat it as a number of pansharpening sub-problems, where each band of the HrMS image plays the role of a panchromatic image. There are mainly two categories of pansharpening methods: component substitution (CS) [5, 17, 1] and multiresolution analysis (MRA) [20, 21, 4, 33, 6]. These methods always suffer from the high spectral distortion,

since a single panchromatic image contains little spectral information as compared with the expected HS image.

In the last few years, machine learning based methods have gained much attention on MS/HS fusion problem [54, 2, 11, 14, 52, 48, 26, 40]. Some of these methods used sparse coding technique to learn a dictionary on the patches across a HrMS image, which delivers spatial knowledge of HrHS to a certain extent, and then learn a coefficient matrix from LrHS to fully represent the HrHS [54, 2, 11, 40]. Some other methods, such as [14], use the sparse matrix factorization to learn a spectral dictionary for LrHS images and then construct HrMS images by exploiting both the spectral dictionary and HrMS images. The low-rankness of HS images can also be exploited with non-negative matrix factorization, which helps to reduce spectral distortions and enhances the MS/HS fusion performance [52, 48, 26]. The main drawback of these methods is that they are mainly designed based on human observations and strong prior assumptions, which may not be very accurate and would not always hold for diverse real world images.

2.2. Deep learning based methods

Recently, a number of DL-based pansharpening methods were proposed by exploiting different network structures [15, 22, 42, 43, 29, 30, 32]. These methods can be easily adapted to MS/HS fusion problem. For example, very recently, [28] proposed a 3D-CNN based MS/HS fusion method by using PCA to reduce the computational cost. This method is usually trained with prepared training data. The network inputs are set as the combination of HrMS/panchromatic images and LrHS/multispectral images (which is usually interpolated to the same spatial size as HrMS/panchromatic images in advance), and the outputs are the corresponding HrHS images. The current DL-based methods have been verified to be able to attain good performance. They, however, just employ networks assembled with some off-the-shelf components in current deep learning toolkits, which are not specifically designed against the investigated problem. Thus the main drawback of this technique is the lack of interpretability to this particular MS/HS fusion task. In specific, both the intrinsic observation model (1), (2) and the evident prior structures, like the spectral correlation property, possessed by HS images have been neglected by such kinds of "black-box" deep model.

3. MS/HS fusion model

In this section, we demonstrate the proposed MS/HS fusion model in detail.

3.1. Model formulation

We first introduce an equivalent formulation for observation model (1). Specifically, we have following theorem[2].

[2]All proofs are presented in supplementary material.

Theorem 1. *For any* $\boldsymbol{X} \in \mathbb{R}^{HW\times S}$ *and* $\tilde{\boldsymbol{Y}} \in \mathbb{R}^{HW\times s}$, *if* $\text{rank}(\boldsymbol{X}) = r > s$ *and* $\text{rank}(\tilde{\boldsymbol{Y}}) = s$, *then the following two statements are equivalent to each other:*
(a) There exists an $\boldsymbol{R} \in \mathbb{R}^{S\times s}$, *subject to,*

$$\tilde{\boldsymbol{Y}} = \boldsymbol{X}\boldsymbol{R}. \tag{3}$$

(b) There exist $\boldsymbol{A} \in \mathbb{R}^{s\times S}$, $\boldsymbol{B} \in \mathbb{R}^{(r-s)\times S}$ *and* $\hat{\boldsymbol{Y}} \in \mathbb{R}^{HW\times(r-s)}$, *subject to,*

$$\boldsymbol{X} = \tilde{\boldsymbol{Y}}\boldsymbol{A} + \hat{\boldsymbol{Y}}\boldsymbol{B}. \tag{4}$$

In reality, the band number of an HrMS image is usually not large, which makes it full rank along spectral mode. For example, the most commonly used HrMS images, RGB images, contain three bands, and their rank along the spectral mode is usually also three. Thus, by letting $\tilde{\boldsymbol{Y}} = \boldsymbol{Y} - \boldsymbol{N}_y$ where $\boldsymbol{Y}$ is the observed HrMS in (1), it is easy to find that $\tilde{\boldsymbol{Y}}$ and $\boldsymbol{X}$ satisfy the conditions in **Theorem 1**. Then the observation model (1) is equivalent[3] to

$$\boldsymbol{X} = \boldsymbol{Y}\boldsymbol{A} + \hat{\boldsymbol{Y}}\boldsymbol{B} + \boldsymbol{N}_x, \tag{5}$$

where $\boldsymbol{N}_x = -\boldsymbol{N}_y\boldsymbol{A}$ is caused by the noise contained in the HrMS image. In (5), $[\boldsymbol{Y}, \hat{\boldsymbol{Y}}]$ can be viewed as r bases that represent columns in $\boldsymbol{X}$ with coefficients matrix $[\boldsymbol{A}; \boldsymbol{B}] \in \mathbb{R}^{r\times S}$, where only the $r - s$ bases in $\hat{\boldsymbol{Y}}$ are unknown. In addition, we can derive the following corollary:

Corollary 1. *For any* $\tilde{\boldsymbol{Y}} \in \mathbb{R}^{HW\times s}$, $\tilde{\boldsymbol{Z}} \in \mathbb{R}^{hw\times S}$, $\boldsymbol{C} \in R^{hw\times HW}$, *if* $\text{rank}(\tilde{\boldsymbol{Y}}) = s$ *and* $\text{rank}(\tilde{\boldsymbol{Z}}) = r > s$, *then the following two statements are equivalent to each other:*
(a) There exist $\boldsymbol{X} \in \mathbb{R}^{HW\times S}$ *and* $\boldsymbol{R} \in \mathbb{R}^{S\times s}$, *subject to,*

$$\tilde{\boldsymbol{Y}} = \boldsymbol{X}\boldsymbol{R}, \quad \tilde{\boldsymbol{Z}} = \boldsymbol{C}\boldsymbol{X}, \quad \text{rank}(\boldsymbol{X}) = r. \tag{6}$$

(b) There exist $\boldsymbol{A} \in \mathbb{R}^{s\times S}$, $r > s$, $\boldsymbol{B} \in \mathbb{R}^{(r-s)\times S}$ *and* $\hat{\boldsymbol{Y}} \in \mathbb{R}^{HW\times(r-s)}$, *subject to,*

$$\tilde{\boldsymbol{Z}} = \boldsymbol{C}\left(\tilde{\boldsymbol{Y}}\boldsymbol{A} + \hat{\boldsymbol{Y}}\boldsymbol{B}\right). \tag{7}$$

Let $\tilde{\boldsymbol{Z}} = \boldsymbol{Z} - N_z$, where $\boldsymbol{Z}$ is the observed LrHS image in (2). It is easy to find that, when being viewed as equations of the to-be-estimated $\boldsymbol{X}$, $\boldsymbol{R}$ and $\boldsymbol{C}$, the observation model (1) and model (2) are equivalent to the following equation of $\hat{\boldsymbol{Y}}$, $\boldsymbol{A}$, $\boldsymbol{B}$ and $\boldsymbol{C}$:

$$\boldsymbol{Z} = \boldsymbol{C}\left(\boldsymbol{Y}\boldsymbol{A} + \hat{\boldsymbol{Y}}\boldsymbol{B}\right) + \boldsymbol{N}, \tag{8}$$

where $\boldsymbol{N} = \boldsymbol{N}_z - \boldsymbol{C}\boldsymbol{N}_y\boldsymbol{A}$ denotes the noise contained in HrMS and LrHS image. Then, we can design the following MS/HS fusion model:

$$\min_{\hat{\boldsymbol{Y}}} \left\|\boldsymbol{C}\left(\boldsymbol{Y}\boldsymbol{A} + \hat{\boldsymbol{Y}}\boldsymbol{B}\right) - \boldsymbol{Z}\right\|_F^2 + \lambda f\left(\hat{\boldsymbol{Y}}\right), \tag{9}$$

[3]We say two equation are equivalent to each other if the solution of one equation can easily achieve by solving the other one

where λ is a trade-off parameter, and $f(\cdot)$ is a regularization function. We adopt regularization on the to-be-estimated bases in $\hat{\boldsymbol{Y}}$, rather than on $\boldsymbol{X}$ as in conventional, to facilitate an entire preservation of spatial details[4] contained in the known HrMS bases ($\boldsymbol{Y}$) for representing $\boldsymbol{X}$.

It should be noted that for data obtained with the same sensors, $\boldsymbol{A}$, $\boldsymbol{B}$ and $\boldsymbol{C}$ are fixed. This means that they can be learned from the training data. In the later sections we will show how to learn them with a deep network.

3.2. Model optimization

We now solve (9) using a proximal gradient algorithm [3], which iteratively updates $\hat{\boldsymbol{Y}}$ by calculating

$$\hat{\boldsymbol{Y}}^{(k+1)} = \arg\min_{\hat{\boldsymbol{Y}}} Q\left(\hat{\boldsymbol{Y}}, \hat{\boldsymbol{Y}}^{(k)}\right), \tag{10}$$

where $\hat{\boldsymbol{Y}}^{(k)}$ is the updating result after $k-1$ iterations, $k = 1, 2, \cdots, K$, and $Q(\hat{\boldsymbol{Y}}, \hat{\boldsymbol{Y}}^{(k)})$ is a quadratic approximation [3] defined as:

$$\begin{aligned} Q\left(\hat{\boldsymbol{Y}}, \hat{\boldsymbol{Y}}^{(k)}\right) = & g\left(\hat{\boldsymbol{Y}}^{(k)}\right) + \left\langle \hat{\boldsymbol{Y}} - \hat{\boldsymbol{Y}}^{(k)}, \nabla g\left(\hat{\boldsymbol{Y}}^{(k)}\right) \right\rangle \\ & + \frac{1}{2\eta} \left\| \hat{\boldsymbol{Y}} - \hat{\boldsymbol{Y}}^{(k)} \right\|_F^2 + \lambda f\left(\hat{\boldsymbol{Y}}\right), \end{aligned} \tag{11}$$

where $g(\hat{\boldsymbol{Y}}^{(k)}) = \|\boldsymbol{C}(\boldsymbol{Y}\boldsymbol{A} + \hat{\boldsymbol{Y}}^{(k)}\boldsymbol{B}) - \boldsymbol{Z}\|_F^2$ and η plays the role of stepsize.

It is easy to prove that the problem (10) is equivalent to:

$$\min_{\hat{\boldsymbol{Y}}} \frac{1}{2} \left\| \hat{\boldsymbol{Y}} - \left(\hat{\boldsymbol{Y}}^{(k)} - \eta \nabla g\left(\hat{\boldsymbol{Y}}^{(k)}\right)\right) \right\|_F^2 + \lambda\eta f\left(\hat{\boldsymbol{Y}}\right). \tag{12}$$

For many kinds of regularization terms, the solution of Eq. (12) is usually in closed-form [8], written as:

$$\hat{\boldsymbol{Y}}^{(k+1)} = \text{prox}_{\lambda\eta}\left(\hat{\boldsymbol{Y}}^{(k)} - \eta \nabla g\left(\hat{\boldsymbol{Y}}^{(k)}\right)\right), \tag{13}$$

where $\text{prox}_{\lambda\eta}(\cdot)$ is a proximal operator dependent on $f(\cdot)$. Since $\nabla g(\hat{\boldsymbol{Y}}^{(k)}) = \boldsymbol{C}^T(\boldsymbol{C}(\boldsymbol{Y}\boldsymbol{A} + \hat{\boldsymbol{Y}}^{(k)}\boldsymbol{B}) - \boldsymbol{Z})\boldsymbol{B}^T$, we can obtain the final updating rule for $\hat{\boldsymbol{Y}}$:

$$\hat{\boldsymbol{Y}}^{(k+1)} = \text{prox}_{\lambda\eta}\left(\hat{\boldsymbol{Y}}^{(k)} - \eta \boldsymbol{C}^T\left(\boldsymbol{C}\left(\boldsymbol{Y}\boldsymbol{A} + \hat{\boldsymbol{Y}}^{(k)}\boldsymbol{B}\right) - \boldsymbol{Z}\right)\boldsymbol{B}^T\right). \tag{14}$$

We can then unfold this algorithm into a deep network.

4. MS/HS fusion net

Based on the above algorithm, we build a deep neural network for MS/HS fusion by unfolding all steps of the algorithm as network layers. This technique has been widely utilized in various computer vision tasks and has been substantiated to be effective in compressed sensing, dehazing, deconvolution, etc. [44, 45, 53]. The proposed network is a structure of K stages, corresponding to K iterations in the iterative algorithm for solving Eq. (9), as shown in Fig. 3 (a) and (b). Each stage takes the HrMS image $\boldsymbol{Y}$, LrHS image $\boldsymbol{Z}$, and the output of the previous stage $\hat{\boldsymbol{Y}}$, as inputs, and outputs an updated $\hat{\boldsymbol{Y}}$ to be the input of next layer.

Iterative optimization algorithm	Network design
For $k = 1{:}K$ do:	In stage $k = 1{:}K$ of the network do:
$\boldsymbol{X}^{(k)} = \boldsymbol{Y}\boldsymbol{A} + \hat{\boldsymbol{Y}}^{(k)}\boldsymbol{B}$	$\mathcal{X}^{(k)} = \mathcal{Y} \times_3 \boldsymbol{A}^T + \hat{\mathcal{Y}}^{(k)} \times_3 \boldsymbol{B}^T$
$\boldsymbol{E}^{(k)} = \boldsymbol{C}\boldsymbol{X}^{(k)} - \boldsymbol{Z}$	$\mathcal{E}^{(k)} = \text{downSample}_{\theta_d^{(k)}}(\mathcal{X}^{(k)}) - \mathcal{Z}$
$\boldsymbol{G}^{(k)} = \eta \boldsymbol{C}^T \boldsymbol{E}^{(k)} \boldsymbol{B}^T$	$\mathcal{G}^{(k)} = \eta \cdot \text{upSample}_{\theta_d^{(k)}}(\mathcal{E}^{(k)}) \times_3 \boldsymbol{B}$
$\boldsymbol{Y}^{(k+1)} = \text{prox}_{\lambda\eta}(\hat{\boldsymbol{Y}}^{(k)} - \boldsymbol{G}^{(k)})$	$\hat{\mathcal{Y}}^{(k+1)} = \text{proxNet}_{\theta_p^{(k)}}(\hat{\mathcal{Y}}^{(k)} - \mathcal{G}^{(k)})$

Figure 2. An illustration of relationship between the algorithm with matrix form and the network structure with tensor form.

4.1. Network design

Algorithm unfolding. We first decompose the updating rule (14) into the following equivalent four sequential parts:

$$\boldsymbol{X}^{(k)} = \boldsymbol{Y}\boldsymbol{A} + \hat{\boldsymbol{Y}}^{(k)}\boldsymbol{B}, \tag{15}$$

$$\boldsymbol{E}^{(k)} = \boldsymbol{C}\boldsymbol{X}^{(k)} - \boldsymbol{Z}, \tag{16}$$

$$\boldsymbol{G}^{(k)} = \eta \boldsymbol{C}^T \boldsymbol{E}^{(k)} \boldsymbol{B}^T, \tag{17}$$

$$\hat{\boldsymbol{Y}}^{(k+1)} = \text{prox}_{\lambda\eta}\left(\hat{\boldsymbol{Y}}^{(k)} - \boldsymbol{G}^{(k)}\right). \tag{18}$$

In the network framework, we use the images with their tensor formulations ($\mathcal{X} \in \mathbb{R}^{H\times W\times S}$, $\mathcal{Y} \in \mathbb{R}^{H\times W\times s}$ and $\mathcal{Z} \in \mathbb{R}^{h\times w\times S}$) instead of their matrix forms to protect their spatial structure knowledge and make the network structure (in tensor form) easily designed. We then design a network to approximately perform the above operations in tensor version. Refer to Fig. 2 for easy understanding.

In tensor version, Eq. (15) can be easily performed by the two multiplications between a tensor and a matrix along the 3^{rd} mode of the tensor. Specifically, in the TensorFlow[5] framework, multiplying a tensor with an matrix in $\mathbb{R}^{m\times n}$ along the channel mode can be easily performed by using the 2D convolution function with a associated $1\times 1\times m\times n$ tensor. Thus, we can perform the tensor version of (15) by:

$$\mathcal{X}^{(k)} = \mathcal{Y} \times_3 \boldsymbol{A}^T + \hat{\mathcal{Y}}^{(k)} \times_3 \boldsymbol{B}^T, \tag{19}$$

where $\times_3$ denotes the mode-3 multiplication for tensor[6].

In Eq. (16), the matrix $\boldsymbol{C}$ represents the spatial down-sampling operator, which can be decomposed into 2D convolutions and down-sampling operators [12, 24, 25]. Thus, we perform the tensor version of (16) by:

$$\mathcal{E}^{(k)} = \text{downSample}_{\theta_d^{(k)}}\left(\mathcal{X}^{(k)}\right) - \mathcal{Z}, \tag{20}$$

[4] Directly imposing regularization terms on $\boldsymbol{X}$, e.g., TV norm, will lead to losing of details like the sharp edge and lines in $\boldsymbol{X}$.

[5] https://tensorflow.google.cn/

[6] For a tensor $\mathcal{U} \in \mathbb{R}^{I\times J\times K}$ with u_{ijk} as its elements, and $\boldsymbol{V} \in \mathbb{R}^{K\times L}$ with v_{kl} as its elements, let $\mathcal{W} = \mathcal{U} \times_3 \boldsymbol{V}$, the elements of $\mathcal{W}$ are $w_{ijl} = \sum_{k=1}^{K} u_{ijk} v_{lk}$. Besides, $\mathcal{W} = \mathcal{U} \times_3 \boldsymbol{V} \Leftrightarrow \boldsymbol{W} = \boldsymbol{U}\boldsymbol{V}^T$.

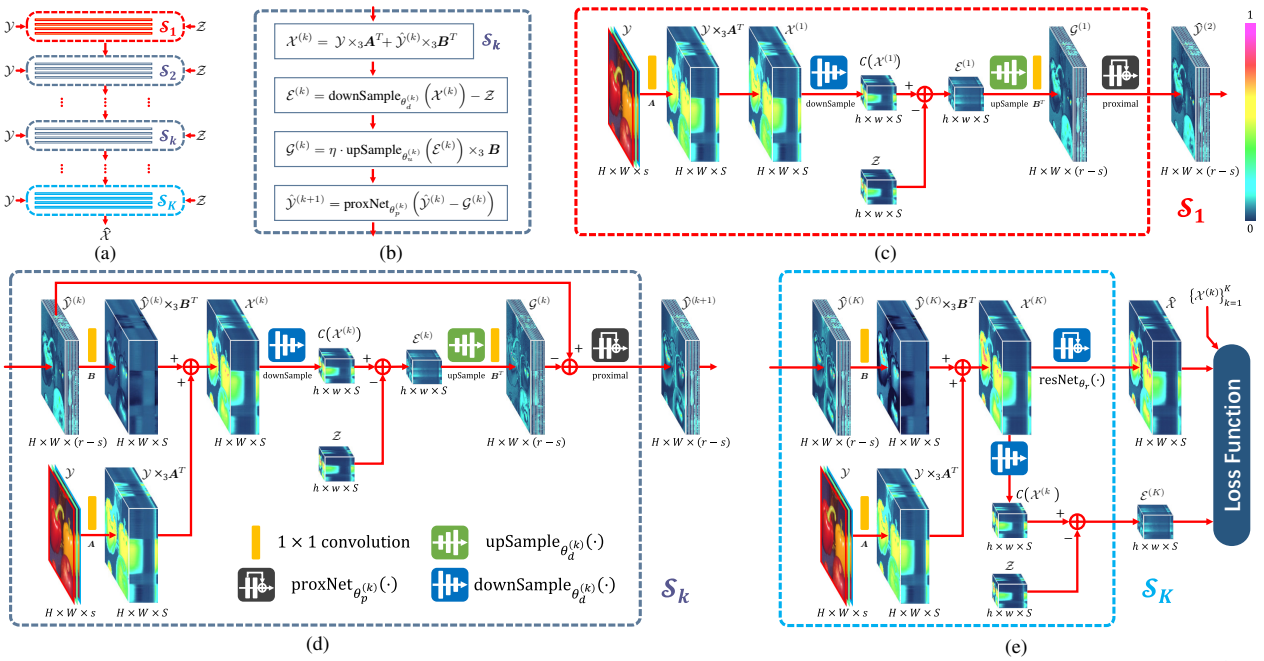


Figure 3. (a) The proposed network with K stages implementing K iterations in the iterative optimization algorithm, where the k^{th} stage is denoted as $\mathcal{S}_k$, $(k = 1, 2, \cdots, K)$. (b) The flowchart of k^{th} $(k < K)$ stage. (c)-(e) Illustration of the first, k^{th} $(1 < k < K)$ and final stage of the proposed network, respectively. When setting $\hat{\mathcal{Y}}^{(k)} = \mathbf{0}$, $\mathcal{S}_k$ is equivalent to $\mathcal{S}_1$.

where $\mathcal{E}^{(k)}$ is an $h \times w \times S$ tensor, downSample$_{\theta_d^{(k)}}(\cdot)$ is the downsampling network consisting of 2D channel-wise convolutions and average pooling operators, and $\theta_d^{(k)}$ denotes filters involved in the operator at the k^{th} stage of network.

In Eq. (17), the transposed matrix $\boldsymbol{C}^T$ represents a spatial upsampling operator. This operator can be easily performed by exploiting the 2D transposed convolution [9], which is the transposition of the downsampling operator. By exploiting the 2D transposed convolution with the filter size as (20), we can approach (17) in the network by:

$$\mathcal{G}^{(k)} = \eta \cdot \text{upSample}_{\theta_u^{(k)}}\left(\mathcal{E}^{(k)}\right) \times_3 \boldsymbol{B}, \tag{21}$$

where $\mathcal{G}^{(k)} \in \mathbb{R}^{H \times W \times S}$, upSample$_{\theta_u^{(k)}}(\cdot)$ is the spacial upsampling network consisting of transposed convolutions and $\theta_u^{(k)}$ denotes the corresponding filters in the k^{th} stage.

In Eq. (18), prox$(\cdot)$ is a to-be-decided proximal operator. We adopt the deep residual network (ResNet) [13] to learn this operator. We then represent (18) in our network as:

$$\hat{\mathcal{Y}}^{(k+1)} = \text{proxNet}_{\theta_p^{(k)}}\left(\hat{\mathcal{Y}}^{(k)} - \mathcal{G}^{(k)}\right), \tag{22}$$

where proxNet$_{\theta_p^{(k)}}(\cdot)$ is a ResNet which represents the proximal operator in our algorithm and the parameters involved in the ResNet at the k^{th} stage are denoted by $\theta_p^{(k)}$.

With Eq. (19)-(22), we can now construct the stages in the proposed network. Fig. 3 (b) shows the flowchart of a single stage of the proposed network.

Normal stage. In the first stage, we simply set $\hat{\mathcal{Y}}^{(1)} = \mathbf{0}$. By exploiting (19)-(22), we can obtain the first network stage as shown in Fig. 3 (c). Fig. 3 (d) shows the k^{th} stage $(1 < k < K)$ of the network obtained by utilizing (19)-(22).

Final stage. As shown in Fig. 3(e), in the final stage, we can approximately generate the HrHS image by (19). Note that $\boldsymbol{X}^{(K)}$ (the unfolding matrix of $\mathcal{X}^{(K)}$) has been intrinsically encoded with low-rank structure. Moreover, according to **Theorem 1**, there exists an $\boldsymbol{R} \in \mathbb{R}^{S \times s}$, s.t., $\boldsymbol{Y} = \boldsymbol{X}^{(K)}\boldsymbol{R}$, which satisfies the observation model (1).

However, HrMS images $\mathcal{Y}$ are usually corrupted with slight noise in reality, and there is a little gap between the low rank assumption and the real situation. This implies that $\boldsymbol{X}^{(K)}$ is not exactly equivalent to the to-be-estimated HrHS image. Therefore, as shown in Fig. 3 (e), in the final stage of the network, we add a ResNet on $\mathcal{X}^{(K)}$ to reduce the gap between the to-be-estimated HrHS image and $\mathcal{X}^{(K)}$:

$$\hat{\mathcal{X}} = \text{resNet}_{\theta_r}\left(\mathcal{X}^{(K)}\right). \tag{23}$$

In this way, we design an end-to-end training architecture, called MS/HS fusion net or MHFnet. We denote the function of entire net as $\hat{\mathcal{X}} = \text{MHFnet}\left(\mathcal{Y}, \mathcal{Z}, \Theta\right)$, where Θ represents all parameters involved in the network. Please refer to supplementary material for more details.

4.2. Network training

Training loss. As shown in Fig. 3 (e), the training loss for each training image is defined as following:

$$L = \|\hat{\mathcal{X}} - \mathcal{X}\|_F^2 + \alpha \sum_{k=1}^{K} \|\mathcal{X}^{(k)} - \mathcal{X}\|_F^2 + \beta \|\mathcal{E}^{(K)}\|_F^2, \tag{24}$$

where $\hat{\mathcal{X}}$ and $\mathcal{X}^{(k)}$ are the final and per-stage outputs of the proposed network, α and β are two trade-off parameters[7].

[7]We set α and β with small values (0.1 and 0.01, respectively) in all experiments, to make the first term play a dominant role.

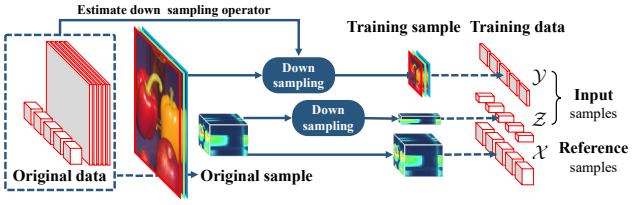


Figure 4. Illustration of how to create the training data when HrHS images are unavailable.

The first term is the pixel-wise L_2 distance between the output of the proposed network and the ground truth $\mathcal{X}$, which is the main component of our loss function. The second term is the pixel-wise L_2 distance between the output $\mathcal{X}^{(k)}$ and the ground truth $\mathcal{X}$ in each stage. This term helps find the correct parameters in each stage, since appropriate $\hat{\mathcal{Y}}^{(k)}$ would lead to $\hat{\mathcal{X}^{(k)}} \approx \mathcal{X}$. The final term is the pixel-wise L_2 distance of the residual of observation model (2) for the final stage of the network.

Training data. For simulation data and real data with available ground-truth HrHS images, we can easily use the paired training data $\{(\mathcal{Y}_n, \mathcal{Z}_n), \mathcal{X}_n\}_{n=1}^N$ to learn the parameters in the proposed MHF-net. Unfortunately, for real data, HrHS images $\mathcal{X}_n$s are sometimes unavailable. In this case, we use the method proposed in [30] to address this problem, where the Wald protocol [50] is used to create the training data as shown in Fig. 4. We downsample both HrMS images and LrHS images in space, so that the original LrHS images can be taken as references for the downsampled data. Please refer to supplementary material for more details.

Implementation details. We implement and train our network using TensorFlow framework. We use Adam optimizer to train the network for 50000 iterations with a batch size of 10 and a learning rate of 0.0001. The initializations of the parameters and other implementation details are listed in supplementary materials.

5. Experimental results

We first conduct simulated experiments to verify the mechanism of MHF-net quantitatively. Then, experimental results on simulated and real data sets are demonstrated to evaluate the performance of MHF-net.

Evaluation measures. Five quantitative picture quality indices (PQI) are employed for performance evaluation, including peak signal-to-noise ratio (PSNR), spectral angle mapper (SAM) [49], erreur relative globale adimensionnelle de synthèse (ERGAS [38]), structure similarity (SSIM [39]), feature similarity (FSIM [51]). SAM calculates the average angle between spectrum vectors of the target MSI and the reference one across all spatial positions and ERGAS measures fidelity of the restored image based on the weighted sum of MSE in each band. PSNR, SSIM and FSIM are conventional PQIs. They evaluate the similarity between the target and the reference images based on MSE and structural consistency, perceptual consistency, respectively. The smaller ERGAS and SAM are, and the larger PSNR, SSIM and FSIM are, the better the fusion result is.

Table 1. Average performance of the competing methods over 12 testing samples of CAVE data set with respect to 5 PQIs.

	ResNet	MHF-net with (K, L)			
		$(4, 9)$	$(7, 5)$	$(10, 4)$	$(13, 2)$
PSNR	32.25	36.15	36.61	36.85	**37.23**
SAM	19.093	9.206	8.636	7.587	**7.298**
ERGA	141.28	92.94	88.56	86.53	**81.87**
SSIM	0.865	0.948	0.955	0.960	**0.962**
FSIM	0.966	0.974	0.975	0.975	**0.976**

5.1. Model verification with CAVE data

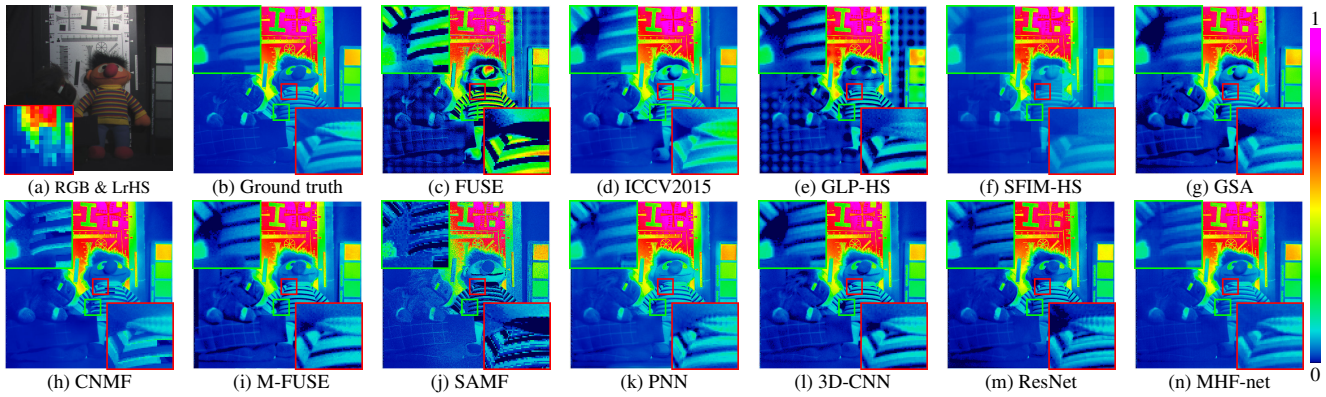
To verify the efficiency of the proposed MHF-net, we first compare the performance of MHF-net with different settings on the CAVE Multispectral Image Database [46][8]. The database consists of 32 scenes with spatial size of 512×512, including full spectral resolution reflectance data from 400nm to 700nm at 10nm steps (31 bands in total). We generate the HrMS image (RGB image) by integrating all the ground truth HrHS bands with the same simulated spectral response $\boldsymbol{R}$, and generate the LrHS images via downsampling the ground-truth with a factor of 32 implemented by averaging over 32×32 pixel blocks as [2, 16].

To prepare samples for training, we randomly select 20 HS images from CAVE database and extract 96×96 overlapped patches from them as reference HrHS images for training. Then the utilized HrHS, HrMS and LrHS images are of size $96 \times 96 \times 31$, $96 \times 96 \times 3$ and $3 \times 3 \times 31$, respectively. The remaining 12 HS images of the database are used for validation, where the original images are treated as ground truth HrHS images, and the HrMS and LrHS images are generated similarly as the training samples.

We compare the performance of the proposed MHF-net under different stage number K. In order to make the competition fair, we adjust the level number L of the ResNet used in proxNet$_{\theta_p^{(k)}}$ for each situation, so that the total level number of the network in each setting is similar to each other. Moreover, to better verify the efficiency of the proposed network, we implement another network for competition, which only uses the ResNet in (22) and (23) without using other structures in MHF-net. This method is simply denoted as "ResNet". In this method, we set the input as $[\mathcal{Y}, \mathcal{Z}_{up}]$, where $\mathcal{Z}_{up}$ is obtained by interpolating the LrHS image $\mathcal{Z}$ (using a bicubic filter) to the dimension of $\mathcal{Y}$ as [28] did. We set the level number of ResNet to be 30.

Table 1 shows the average results over 12 testing HS images of two DL methods in different settings. We can observe that MHF-net with more stages, even with fewer net levels in total, can significantly lead to better performance. We can also observe that the MHF-net can achieve better

[8] http://www.cs.columbia.edu/CAVE/databases/



(a) RGB & LrHS (b) Ground truth (c) FUSE (d) ICCV2015 (e) GLP-HS (f) SFIM-HS (g) GSA

(h) CNMF (i) M-FUSE (j) SAMF (k) PNN (l) 3D-CNN (m) ResNet (n) MHF-net

Figure 5. (a) The simulated RGB (HrMS) and LrHS (left bottom) images of *chart and staffed toy*, where we display the 10th (490nm) band of the HS image. (b) The ground-truth HrHS image. (c)-(n) The results obtained by 12 comparison methods, with two demarcated areas zoomed in 4 times for easy observation.

Table 2. Average performance of the competing methods over 12 testing images of CAVE date set with respect to 5 PQIs.

	PSNR	SAM	ERGAS	SSIM	FSIM
FUSE	30.95	13.07	188.72	0.842	0.933
ICCV15	32.94	10.18	131.94	0.919	0.961
GLP-HS	33.07	11.58	126.04	0.891	0.942
SFIM-HS	31.86	7.63	147.41	0.914	0.932
GSA	33.78	11.56	122.50	0.884	0.959
CNMF	33.59	8.22	122.12	0.929	0.964
M-FUSE	32.11	8.82	151.97	0.914	0.947
SASFM	26.59	11.25	362.70	0.799	0.916
PNN	32.42	14.73	134.51	0.884	0.956
3D-CNN	34.82	8.96	109.20	0.937	0.971
ResNet	32.25	16.14	141.28	0.865	0.966
MHF-net	**37.23**	**7.30**	**81.87**	**0.962**	**0.976**

results than ResNet (about 5db in PSNR), while the main difference between MHF-net and ResNet is our proposed stage structure in the network. These results show that the proposed stage structure in MHF-net, which introduces interpretability specifically to the problem, can indeed help enhance the performance of MS/HS fusion.

5.2. Experiments with simulated data

We then evaluate MHF-net on simulated data in comparison with state-of-art methods.

Comparison methods. The comparison methods include: FUSE [41][9], ICCV15 [18][10], GLP-HS [31][11], SFIM-HS [19][11], GSA [1][11], CNMF [48][12], M-FUSE [40][13] and SASFM [14][14], representing the state-of-the-art traditional methods; PNN [30] and 3D-CNN [28] representing the state-of-the-art DL-based methods. We also compare the proposed MHF-net with the implemented ResNet method.

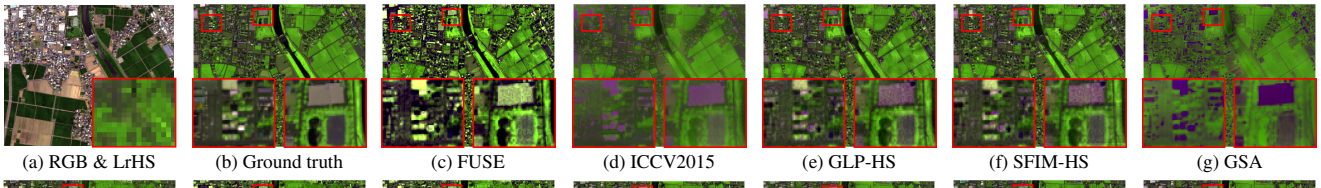
Performance comparison with CAVE data. With the same experiment setting as in the previous section, we compare the performance of all competing methods on the 12 testing HS images ($K = 13$ and $L = 2$ in MHF-net). Table 2 lists the average performance over all testing images of all comparison methods. From the table, it is seen that the proposed MHF-net method can significantly outperform other competing methods with respect to all evaluation measures. Fig. 5 shows the 10-th band (490nm) of the HS image *chart and staffed toy* obtained by the completing methods. It is easy to observe that the proposed method performs better than other competing ones, in the better recovery of both finer-grained textures and coarser-grained structures. More results are depicted in the supplementary material.

Performance comparison with Chikusei data. The Chikusei data set [47][15] is an airborne HS image taken over Chikusei, Ibaraki, Japan, on 29 July 2014. The data set is of size $2517 \times 2335 \times 128$ with the spectral range from 0.36 to 1.018. We view the original data as the HrHS image and simulate the HrMS (RGB image) and LrMS (with a factor of 32) image in the similar way as the previous section.

We select a 500×2210-pixel-size image from the top area of the original data for training, and extract 96×96 overlapped patches from the training data as reference HrHS images for training. The input HrHS, HrMS and LrHS samples are of sizes $96 \times 96 \times 128$, $96 \times 96 \times 3$ and $3 \times 3 \times 128$, respectively. Besides, from remaining part of the original image, we extract 16 non-overlap $448 \times 544 \times 128$ images as testing data. More details about the experimental setting are introduced in supplementary material.

Table 3 shows the average performance over 16 testing images of all competing methods. It is easy to observe that the proposed method significantly outperforms other methods with respect to all evaluation measures. Fig. 6 shows the composite images of a test sample obtained by the competing methods, with bands 70-100-36 as R-G-B. It is seen that the composite image obtained by MHF-net is closest to the ground-truth, while the results of other methods usually contain obvious incorrect structure or spectral distortion. More results are listed in supplementary material.

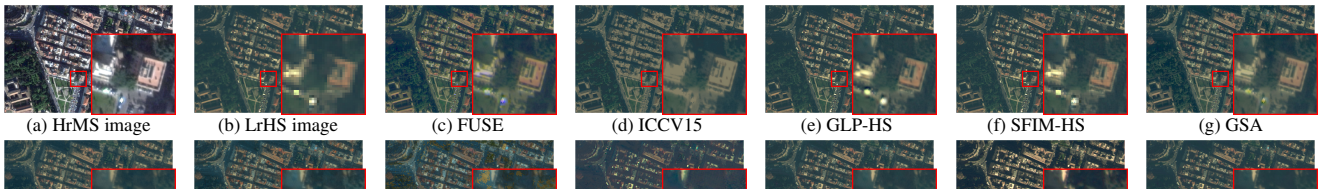
[9] http://wei.perso.enseeiht.fr/publications.html
[10] https://github.com/lanha/SupResPALM
[11] http://openremotesensing.net/knowledgebase/hyperspectral-and-multispectral-data-fusion/
[12] http://naotoyokoya.com/Download.html
[13] https://github.com/qw245/BlindFuse
[14] We write the code by ourselves.
[15] http://naotoyokoya.com/Download.html



(a) RGB & LrHS (b) Ground truth (c) FUSE (d) ICCV2015 (e) GLP-HS (f) SFIM-HS (g) GSA

(h) CNMF (i) M-FUSE (j) SAMF (k) PNN (l) 3D-CNN (m) ResNet (n) MHF-net

Figure 6. (a) The simulated RGB (HrMS) and LrHS (left bottom) images of a test sample in Chikusei data set. We show the composite image of the HS image with bands 70-100-36 as R-G-B. (b) The ground-truth HrHS image. (c)-(n) The results obtained by 10 comparison methods, with two demarcated areas zoomed in 4 times for easy observation.



(a) HrMS image (b) LrHS image (c) FUSE (d) ICCV15 (e) GLP-HS (f) SFIM-HS (g) GSA

(h) CNMF (i) M-FUSE (j) SAMF (k) PNN (l) 3D-CNN (m) ResNet (n) MHF-net

Figure 7. (a) and (b) are the HrMS (RGB) and LrHS images of the left bottom area of *Roman Colosseum* acquired by World View-2 (WV-2). We show the composite image of the HS image with bands 5-3-2 as R-G-B. (c)-(n) The results obtained by 10 comparison methods, with a demarcated area zoomed in 5 times for easy observation.

Table 3. Average performance of the competing methods over 16 testing samples of Chikusei data set with respect to 5 PQIs.

	PSNR	SAM	ERGAS	SSIM	FSIM
FUSE	26.59	7.92	272.43	0.718	0.860
ICCV15	27.77	3.98	178.14	0.779	0.870
GLP-HS	28.85	4.17	163.60	0.796	0.903
SFIM-HS	28.50	4.22	167.85	0.793	0.900
GSA	27.08	5.39	238.63	0.673	0.835
CNMF	28.78	3.84	173.41	0.780	0.898
M-FUSE	24.85	6.62	282.02	0.642	0.849
SASFM	24.93	7.95	369.35	0.636	0.845
PNN	24.30	4.26	157.49	0.717	0.807
3D-CNN	30.51	**3.02**	129.11	0.869	0.933
ResNet	29.35	3.69	144.12	0.866	0.930
MHF-net	**32.26**	**3.02**	**109.55**	**0.890**	**0.946**

5.3. Experiments with real data

In this section, sample images of *Roman Colosseum* acquired by World View-2 (WV-2) are used in our experiments[16]. This data set contains an HrMS image (RGB image) of size $1676 \times 2632 \times 3$ and an LrHS image of size $419 \times 658 \times 8$, while the HrHS image is not available. We select the top half part of the HrMS ($836 \times 2632 \times 3$) and LrHS ($209 \times 658 \times 8$) image to train the MHF-net, and exploit the remaining parts of the data set as testing data. We first extract the training data into $144 \times 144 \times 3$ overlapped HrMS patches and $36 \times 36 \times 3$ overlapped LrHS patches and then generate the training samples by the method as shown in Fig. 4. The input HrHS, HrMS and LrHS samples are of size $36 \times 36 \times 8$, $36 \times 36 \times 3$ and $9 \times 9 \times 8$, respectively.

Fig. 6 shows a portion of the fusion result of the testing data (left bottom area of the original image). Visual inspection evidently shows that the proposed method gives the better visual effect. By comparing with the results of ResNet, we can find that the results of both methods are clear, but the color and brightness of result of the proposed method are much closer to the LrHS image.

[16] https://www.harrisgeospatial.com/DataImagery/SatelliteImagery/HighResolution/WorldView-2.aspx

6. Conclusion

In this paper, we have provided a new MS/HS fusion network. The network takes the advantage of deep learning that all parameters can be learned from the training data with fewer prior pre-assumption on the data, and furthermore takes into account the generation mechanism underlying the MS/HS fusion data. This is achieved by constructing a new MS/HS fusion model based on the observation models, and unfolding the algorithm into an optimization-inspired deep network. The network is thus specifically interpretable to the task, and can help discover the spatial and spectral response operators in a purely end-to-end manner. Experiments implemented on simulated and real MS/HS fusion cases have substantiated the superiority of the proposed MHF-net over the state-of-the-art methods.

Acknowledgment. This research was supported by National Key R&D Program of China (2018YFB1004300) and China NSFC projects (61661166011, 11690011, 61603292, 61721002, U1811461, 61671182)

References

[1] B. Aiazzi, S. Baronti, and M. Selva. Improving component substitution pansharpening through multivariate regression of ms + pan data. *IEEE Transactions on Geoscience and Remote Sensing*, 45(10):3230–3239, 2007. 2, 7

[2] N. Akhtar, F. Shafait, and A. Mian. Sparse spatio-spectral representation for hyperspectral image super-resolution. In *European Conference on Computer Vision*, pages 63–78. Springer, 2014. 2, 3, 6

[3] A. Beck and M. Teboulle. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM journal on imaging sciences*, 2(1):183–202, 2009. 2, 4

[4] P. J. Burt and E. H. Adelson. The laplacian pyramid as a compact image code. In *Readings in Computer Vision*, pages 671–679. Elsevier, 1987. 2

[5] P. Chavez, S. C. Sides, J. A. Anderson, et al. Comparison of three different methods to merge multiresolution and multispectral data- landsat tm and spot panchromatic. *Photogrammetric Engineering and remote sensing*, 57(3):295–303, 1991. 2

[6] M. N. Do and M. Vetterli. The contourlet transform: an efficient directional multiresolution image representation. *IEEE Transactions on image processing*, 14(12):2091–2106, 2005. 2

[7] C. Dong, C. C. Loy, K. He, and X. Tang. Image super-resolution using deep convolutional networks. *IEEE transactions on pattern analysis and machine intelligence*, 38(2):295–307, 2016. 2

[8] D. L. Donoho. De-noising by soft-thresholding. *IEEE transactions on information theory*, 41(3):613–627, 1995. 4

[9] V. Dumoulin and F. Visin. A guide to convolution arithmetic for deep learning. *arXiv preprint arXiv:1603.07285*, 2016. 5

[10] M. Fauvel, Y. Tarabalka, J. A. Benediktsson, J. Chanussot, and J. C. Tilton. Advances in spectral-spatial classification of hyperspectral images. *Proceedings of the IEEE*, 101(3):652–675, 2013. 1

[11] C. Grohnfeldt, X. Zhu, and R. Bamler. Jointly sparse fusion of hyperspectral and multispectral imagery. In *IGARSS*, pages 4090–4093, 2013. 2, 3

[12] R. C. Hardie, M. T. Eismann, and G. L. Wilson. Map estimation for hyperspectral image resolution enhancement using an auxiliary sensor. *IEEE Transactions on Image Processing*, 13(9):1174–1184, 2004. 1, 4

[13] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 5

[14] B. Huang, H. Song, H. Cui, J. Peng, and Z. Xu. Spatial and spectral image fusion using sparse matrix factorization. *IEEE Transactions on Geoscience and Remote Sensing*, 52(3):1693–1704, 2014. 2, 3, 7

[15] W. Huang, L. Xiao, Z. Wei, H. Liu, and S. Tang. A new pan-sharpening method with deep neural networks. *IEEE Geoscience and Remote Sensing Letters*, 12(5):1037–1041, 2015. 3

[16] R. Kawakami, Y. Matsushita, J. Wright, M. Ben-Ezra, Y.-W. Tai, and K. Ikeuchi. High-resolution hyperspectral imaging via matrix factorization. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pages 2329–2336. IEEE, 2011. 6

[17] C. A. Laben and B. V. Brower. Process for enhancing the spatial resolution of multispectral imagery using pan-sharpening, Jan. 4 2000. US Patent 6,011,875. 2

[18] C. Lanaras, E. Baltsavias, and K. Schindler. Hyperspectral super-resolution by coupled spectral unmixing. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3586–3594, 2015. 7

[19] J. Liu. Smoothing filter-based intensity modulation: A spectral preserve image fusion technique for improving spatial details. *International Journal of Remote Sensing*, 21(18):3461–3472, 2000. 7

[20] L. Loncan, L. B. Almeida, J. M. Bioucas-Dias, X. Briottet, J. Chanussot, N. Dobigeon, S. Fabre, W. Liao, G. A. Licciardi, M. Simoes, et al. Hyperspectral pansharpening: A review. *arXiv preprint arXiv:1504.04531*, 2015. 2

[21] S. G. Mallat. A theory for multiresolution signal decomposition: the wavelet representation. *IEEE transactions on pattern analysis and machine intelligence*, 11(7):674–693, 1989. 2

[22] G. Masi, D. Cozzolino, L. Verdoliva, and G. Scarpa. Pansharpening by convolutional neural networks. *Remote Sensing*, 8(7):594, 2016. 3

[23] S. Michel, M.-J. LEFEVRE-FONOLLOSA, and S. HOSFORD. Hypxim–a hyperspectral satellite defined for science, security and defence users. *PAN*, 400(800):400, 2011. 1

[24] R. Molina, A. K. Katsaggelos, and J. Mateos. Bayesian and regularization methods for hyperparameter estimation in image restoration. *IEEE Transactions on Image Processing*, 8(2):231–246, 1999. 1, 2, 4

[25] R. Molina, M. Vega, J. Mateos, and A. K. Katsaggelos. Variational posterior distribution approximation in bayesian super resolution reconstruction of multispectral images. *Applied and Computational Harmonic Analysis*, 24(2):251–267, 2008. 1, 2, 4

[26] Z. H. Nezhad, A. Karami, R. Heylen, and P. Scheunders. Fusion of hyperspectral and multispectral images using spectral unmixing and sparse coding. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 9(6):2377–2389, 2016. 2, 3

[27] F. Palsson, J. R. Sveinsson, and M. O. Ulfarsson. A new pansharpening algorithm based on total variation. *IEEE Geoscience and Remote Sensing Letters*, 11(1):318–322, 2014. 2

[28] F. Palsson, J. R. Sveinsson, and M. O. Ulfarsson. Multispectral and hyperspectral image fusion using a 3-D-convolutional neural network. *IEEE Geoscience and Remote Sensing Letters*, 14(5):639–643, 2017. 2, 3, 6, 7

[29] Y. Rao, L. He, and J. Zhu. A residual convolutional neural network for pan-shaprening. In *Remote Sensing with Intelligent Processing (RSIP), 2017 International Workshop on*, pages 1–4. IEEE, 2017. 3

[30] G. Scarpa, S. Vitale, and D. Cozzolino. Target-adaptive cnn-based pansharpening. *IEEE Transactions on Geoscience and Remote Sensing*, (99):1–15, 2018. 2, 3, 6, 7

[31] M. Selva, B. Aiazzi, F. Butera, L. Chiarantini, and S. Baronti. Hyper-sharpening: A first approach on sim-ga data. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 8(6):3008–3024, 2015. 7

[32] Z. Shao and J. Cai. Remote sensing image fusion with deep convolutional neural network. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 11(5):1656–1669, 2018. 3

[33] J.-L. Starck, J. Fadili, and F. Murtagh. The undecimated wavelet decomposition and its reconstruction. *IEEE Transactions on Image Processing*, 16(2):297–309, 2007. 2

[34] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1–9, 2015. 2

[35] Y. Tarabalka, J. Chanussot, and J. A. Benediktsson. Segmentation and classification of hyperspectral images using minimum spanning forest grown from automatically selected markers. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 40(5):1267–1279, 2010. 1

[36] M. Uzair, A. Mahmood, and A. S. Mian. Hyperspectral face recognition using 3d-dct and partial least squares. In *BMVC*, 2013. 1

[37] H. Van Nguyen, A. Banerjee, and R. Chellappa. Tracking via object reflectance using a hyperspectral video camera. In *Computer Vision and Pattern Recognition Workshops (CVPRW), 2010 IEEE Computer Society Conference on*, pages 44–51. IEEE, 2010. 1

[38] L. Wald. *Data Fusion: Definitions and Architectures: Fusion of Images of Different Spatial Resolutions*. Presses des lEcole MINES, 2002. 6

[39] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE Trans. Image Processing*, 13(4):600–612, 2004. 6

[40] Q. Wei, J. Bioucas-Dias, N. Dobigeon, J.-Y. Tourneret, and S. Godsill. Blind model-based fusion of multi-band and panchromatic images. In *Multisensor Fusion and Integration for Intelligent Systems (MFI), 2016 IEEE International Conference on*, pages 21–25. IEEE, 2016. 2, 3, 7

[41] Q. Wei, N. Dobigeon, and J.-Y. Tourneret. Fast fusion of multi-band images based on solving a sylvester equation. *IEEE Transactions on Image Processing*, 24(11):4109–4121, 2015. 7

[42] Y. Wei and Q. Yuan. Deep residual learning for remote sensed imagery pansharpening. In *Remote Sensing with Intelligent Processing (RSIP), 2017 International Workshop on*, pages 1–4. IEEE, 2017. 3

[43] 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*, 14(10):1795–1799, 2017. 3

[44] D. Yang and J. Sun. Proximal dehaze-net: A prior learning-based deep network for single image dehazing. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 702–717, 2018. 4

[45] Y. Yang, J. Sun, H. Li, and Z. Xu. Admm-net: A deep learning approach for compressive sensing mri. *arXiv preprint arXiv:1705.06869*, 2017. 4

[46] F. Yasuma, T. Mitsunaga, D. Iso, and S. K. Nayar. Generalized assorted pixel camera: postcapture control of resolution, dynamic range, and spectrum. *IEEE transactions on image processing*, 19(9):2241–2253, 2010. 6

[47] N. Yokoya, C. Grohnfeldt, and J. Chanussot. Hyperspectral and multispectral data fusion: A comparative review of the recent literature. *IEEE Geoscience and Remote Sensing Magazine*, 5(2):29–56, 2017. 1, 7

[48] N. Yokoya, T. Yairi, and A. Iwasaki. Coupled non-negative matrix factorization (CNMF) for hyperspectral and multispectral data fusion: Application to pasture classification. In *Geoscience and Remote Sensing Symposium (IGARSS), 2011 IEEE International*, pages 1779–1782. IEEE, 2011. 3, 7

[49] R. H. Yuhas, J. W. Boardman, and A. F. Goetz. Determination of semi-arid landscape endmembers and seasonal trends using convex geometry spectral unmixing techniques. 1993. 6

[50] Y. Zeng, W. Huang, M. Liu, H. Zhang, and B. Zou. Fusion of satellite images in urban area: Assessing the quality of resulting images. In *Geoinformatics, 2010 18th International Conference on*, pages 1–4. IEEE, 2010. 6

[51] L. Zhang, L. Zhang, X. Mou, and D. Zhang. Fsim: a feature similarity index for image quality assessment. *IEEE Trans. Image Processing*, 20(8):2378–2386, 2011. 6

[52] Y. Zhang, Y. Wang, Y. Liu, C. Zhang, M. He, and S. Mei. Hyperspectral and multispectral image fusion using CNMF with minimum endmember simplex volume and abundance sparsity constraints. In *Geoscience and Remote Sensing Symposium (IGARSS), 2015 IEEE International*, pages 1929–1932. IEEE, 2015. 2, 3

[53] J. Zhang13, J. Pan, W.-S. Lai, R. W. Lau, and M.-H. Yang. Learning fully convolutional networks for iterative non-blind deconvolution. 2017. 4

[54] Y. Zhao, J. Yang, Q. Zhang, L. Song, Y. Cheng, and Q. Pan. Hyperspectral imagery super-resolution by sparse representation and spectral regularization. *EURASIP Journal on Advances in Signal Processing*, 2011(1):87, 2011. 2, 3