Patch-Aware Deep Hyperspectral and Multispectral Image Fusion by Unfolding Subspace-Based Optimization Model

Jianjun Liu, *Member, IEEE*, Dunbin Shen, Zebin Wu, *Senior Member, IEEE*, Liang Xiao, *Member, IEEE*, Jun Sun, *Member, IEEE*, and Hong Yan, *Fellow, IEEE*

***Abstract*—Hyperspectral and multispectral image fusion aims to fuse a low-spatial-resolution hyperspectral image (HSI) and a high-spatial-resolution multispectral image to form a high-spatial-resolution HSI. Motivated by the success of model- and deep learning-based approaches, we propose a novel patch-aware deep fusion approach for HSI by unfolding a subspace-based optimization model, where moderate-sized patches are used in both training and test phases. The goal of this approach is to make full use of the information of patch under subspace representation, restrict the scale and enhance the interpretability of the deep network, thereby improving the fusion. First, a subspace-based fusion model was built with two regularization terms to localize pixels and extract texture. Then, the subspace-based fusion model was solved by the alternating direction method of multipliers algorithm, and the model was divided into one fidelity-based problem and two regularization-based problems. Finally, a structured deep fusion network was proposed by unfolding all steps of the algorithm as network layers. Specifically, the fidelity-based problem was solved by a gradient descent algorithm and implemented by a network. The two regularization-based problems were described by proximal operators and learnt by two u-shaped architectures. Moreover, an *aggregation fusion* technique was proposed to improve the performance by averaging the fused images in all iterations and aggregating the overlapping patches in the test phase. Experimental results, conducted on both synthetic and real datasets, demonstrated the effectiveness of the proposed approach.**

Manuscript received October 6, 2021; revised December 2, 2021; accepted December 31, 2021. Date of publication January 5, 2022; date of current version January 20, 2022. This work was supported in part by the National Natural Science Foundation of China under Grant 62071204, Grant 61871226, and Grant 61772274, in part by the Natural Science Foundation of Jiangsu Province under Grant BK20201338 and Grant BK20180018, in part by the Jiangsu Provincial Social Developing Project under Grant BE2018727, in part by the China Postdoctoral Science Foundation under Grant 2021M691275, in part by the Jiangsu Postdoctoral Research Funding Program under Grant 2021K148B, in part by the 111 project under Grant B12018, in part by the Hong Kong Innovation and Technology Commission, and in part by the Hong Kong Research Grants Council under Project CityU 11204821. *(Corresponding author: Jianjun Liu.)*

Jianjun Liu is with the Jiangsu Provincial Engineering Laboratory for Pattern Recognition and Computational Intelligence, School of Artificial Intelligence and Computer Science, Jiangnan University, Wuxi 214126, China, and also with the Department of Electrical Engineering, City University of Hong Kong, Hong Kong (e-mail: liuofficial@163.com).

Dunbin Shen and Jun Sun are with the School of Artificial Intelligence and Computer Science, Jiangnan University, Wuxi 214126, China (e-mail: sdb_2012@163.com; junsun@jiangnan.edu.cn).

Zebin Wu and Liang Xiao are with the School of Computer Science, Nanjing University of Science and Technology, Nanjing 210094, China (e-mail: zebin.wu@gmail.com; xiaoliang@mail.njust.edu.cn).

Hong Yan is with the Department of Electrical Engineering, City University of Hong Kong, Hong Kong (e-mail: h.yan@cityu.edu.hk).

Digital Object Identifier 10.1109/JSTARS.2022.3140211

***Index Terms*—Alternating direction method of multipliers (ADMM), deep learning, hyperspectral image (HSI), image fusion, subspace, unfolding.**

I. Introduction

A HYPERSPECTRAL image (HSI) can be regarded as a 3-D image that contains both spatial and spectral information. For the spectral dimension, HSI is a concatenation of images taken at different spectral bands, with its spectral range covering hundreds of contiguous and narrow bands that span the visible to infrared spectrum. The high spectral resolution of HSIs allows development of applications, such as object detection [1], tracking [2], face recognition [3], [4], and land-cover classification [5]–[8]. Due to the limitations of existing imaging sensors, there is a critical tradeoff between spatial and spectral resolutions [9]. HSIs are acquired with low spatial resolution to ensure high spectral resolution. Conventional multispectral images (MSIs) at much lower spectral resolution can be acquired with higher spatial resolution. Therefore, combining a low-spatial-resolution HSI (LR-HSI) and a high-spatial-resolution MSI (HR-MSI) could be an economical solution to obtain a high-spatial-resolution HSI (HR-HSI) [9]–[11].

The fusion of LR-HSI and HR-MSI, known as hyperspectral and multispectral (HS/MS) image fusion, has attracted great attention [9]–[11]. This problem can be regarded as an extension of the Pansharpening problem that fuses a low-spatial-resolution MSI with a high-spatial-resolution panchromatic image [12]–[15]. Generally, HS/MS image fusion is more complex than that of Pansharpening, and the conventional approaches proposed for Pansharpening can be extended to solve HS/MS image fusion. Related approaches can be divided into four categories—component substitution [16], multiresolution analysis [17], model-based approaches [18]–[35], and deep learning-based approaches [14], [36]–[55]. Among these categories, model- and deep learning-based approaches have been most active recently.

Model-based approaches usually build an optimization model based on the dependence between the target image and the two observed images. The goals are to design effective fidelity terms and exploit efficient regularization terms to obtain the desired result. These models are flexible, and the theory is relatively complete. However, the entire process relies too much

on human experience and there are many empirical parameters that need to be tuned when using prior knowledge to build the optimization model.

Deep learning-based approaches are data-driven and build end-to-end networks to establish the mapping relationship of HS/MS image fusion. These approaches are suitable to exploit the relationship between the given and target data. Unlike the model-based approaches, these approaches rarely consider prior knowledge of the data and their network structures are usually uninterpretable.

Building on the success of model- and deep learning-based approaches, we propose a patch-aware deep fusion network, called SpfNet (i.e., subspace-level fusion network), for HSI by taking advantage of the subspace representation[1] and deep unfolding technique [56]–[64]. First, to make each pixel aware of the pixels around it, singular value decomposition (SVD) is performed on each patch of LR-HSI and a subspace-based optimization model for HS/MS image fusion is built with two regularization terms[2] representing pixel localization and texture extraction introduced to enforce the desired result. Then, the subspace-based model is solved by the alternating direction method of multipliers (ADMM) algorithm and thereby decoupled into three suboptimization problems, one fidelity-based problem and two regularization-based problems. The fidelity-based problem, accounting for spatial-level data fusion, is solved by a gradient descent algorithm, and the regularization-based problems, accounting for pixel localization and texture extraction, are described by proximal operators. Finally, a structured deep network for HS/MS image fusion is constructed by unfolding the iterative algorithm, where the basic calculations are represented as network layers and the proximal operators are replaced by two u-shaped architectures. Moreover, an *aggregation fusion* technique is proposed to improve the quality of HS/MS image fusion. Specifically, the fused images produced in all iterations are convolved and averaged and the overlapping patches are aggregated in the test phase.

Compared with existing HS/MS image fusion approaches, the four innovative characteristics of SpfNet are the following.

1) An interpretable patch-aware deep fusion network is formed by unfolding the subspace-based optimization model. With the network's interpretability, it is easy to adjust the structure according to the physical meaning, and some nonfunctional and redundant parts can be effectively removed.
2) With subspace representation, the channels of input tensors can be restricted to focus on injecting spatial information. A further spectral-spatial fusion is performed to compensate for the loss of information brought by SVD. And it is done by a strategy of image averaging that convolves and averages the fused images produced in all stages.
3) By performing SVD on patches, the coefficients of each pixel can be aware of the full information of its patch. In the test phase, the test images are divided into overlapping patches. The generated pixels that belong to different patches will have different information, and one can make full use of this kind of redundant information by aggregating the overlapping patches.
4) When constructing the network, the fixed format of fidelity terms is broken by a concatenation operator to provide more flexibility, and twin U-nets are proposed for pixel localization and texture extraction.

The rest of this article is organized as follows. Section II briefly reviews related works on HS/MS image fusion. In Section III, the proposed subspace-based optimization model and its ADMM algorithm are introduced and SpfNet is formed by unfolding the ADMM iterations. In Section IV, the effectiveness of SpfNet is demonstrated through experiments on three synthetic datasets and one real dataset. Finally, Section V concludes this article.

II. Related Work

In this section, we briefly review the model-based and deep learning-based approaches to HS/MS image fusion.

A. Model-Based Approaches

Model-based approaches can be roughly divided into two categories: nonfactorization-based and factorization-based approaches. Nonfactorization-based approaches recover the target image entirely and exploit related prior knowledge to enforce the desired result. For example, by a variational Pansharpening model where a priori knowledge of piecewise smooth functions is exploited to regularize the solution [18], or by solving the original unified model by incorporating a sparse tensor representation, where nonlocal similar patches are formulated as tensors [19]. Factorization-based approaches mainly separate the target image into two parts and regenerate it via the recovered parts. The target solution usually has a lower degree of freedom and the computational load is lighter than in nonfactorization-based approaches. There are many strategies proposed for matrix factorization, by making assumptions about the target image. Examples are, that it can be sparsely represented by an overcomplete spectral dictionary and different priors can be used to obtain the spectral dictionary and coefficients [20], [21], [27], [34]; or that it can be represented linearly by pure spectral signatures and the endmember and abundance matrices can be recovered simultaneously [22]–[24]; or that it lives in a low-dimensional subspace and the subspace-based optimization problem can be solved by exploiting prior knowledge, such as piecewise smooth functions [25], dictionary learning [26], tensor multirank [28], truncated matrix decomposition [29], and deep prior [30]; or that it separates the target image into multiple parts by tensor decomposition and updates each part iteratively [31]–[33].

B. Deep Learning-Based Approaches

Learning-based approaches often build a deep network to describe the fusion process, and produce the target image by feeding observed images into the network [11], [14], [36].

[1]In this work, subspace representation is manifested in each patch, which is different from the existing methods that perform the decomposition on the entire image.

[2]Both regularization terms are proposed for preserving spatial information, with pixel localization for pixel-level information and texture extraction for subpixel-level information.

Some approaches enhance the ability to fuse images in the network structures, such as 3-D convolutional neural networks (CNN) [38], residual networks [39], multiscale structures [40], pyramid networks [41], attention networks [42], [43], cross-mode information [44], dense networks [45], [46], adversarial network [47], [48]. Others use detail information from high-spatial-resolution conventional images to improve performance [49]–[52], while some form a hybrid of model- and deep learning-based approaches [53]–[55], [65], [66].

To enhance the interpretability of deep learning-based approaches, constructing a structured deep network by deep unfolding of the iterative algorithm has been used [56]–[64], [67], [68], and the iterative algorithms used include ADMM [58], [59], [68], projected gradient descent [60], proximal gradient [61], [67], half quadratic splitting [62], [63], and iterative shrinkage thresholding [64]. Approaches to HS/MS image fusion include a concise fusion model incorporating a linear representation of the target image followed by a projected gradient method to solve the model with a deep network constructed by unfolding the corresponding iterative algorithm [61]; an iterative formula for HS/MS image fusion according to an observation with detailed compensation processes leading to construction of a structured deep network by unfolding the iterative formula [69]; taking the original unified optimization model with two fidelity terms and one regularization term and splitting the model into three suboptimization problems via a half quadratic splitting algorithm then using a recursive residual network for the subproblem associated with the regularization term and unfolding the two subproblems associated with the fidelity terms into network representations [63].

III. Proposed Approach

A. Subspace-Based Optimization Model

Given two images (i.e., moderate-sized patches) taken from the same scene, a LR-HSI $\mathbf{Y} \in \mathbb{R}^{N_B \times N_w N_h}$ and a HR-MSI $\mathbf{Z} \in \mathbb{R}^{N_b \times N_W N_H}$, HS/MS image fusion aims to generate a HR-HSI $\mathbf{X} \in \mathbb{R}^{N_B \times N_W N_H}$, where N_B and N_b ($N_B > N_b$) are the spectral band numbers, N_W and N_w ($N_W > N_w$) are the spatial widths, and N_H and N_h ($N_H > N_h$) are the spatial heights. We assume that $N_W = rN_w$ and $N_H = rN_h$, where r is the resolution ratio. Some observation models are proposed for $\mathbf{Y}$, $\mathbf{Z}$, and the desired $\mathbf{X}$ [70]. We use the models described as follows:

$$\mathbf{Y} = \mathbf{XB} + \mathbf{E}_Y \tag{1}$$

$$\mathbf{Z} = \mathbf{RX} + \mathbf{E}_Z \tag{2}$$

where $\mathbf{B} \in \mathbb{R}^{N_W N_H \times N_w N_h}$ represents the spatial blur and downsampling, $\mathbf{R} \in \mathbb{R}^{N_b \times N_B}$ represents the spectral response function of the multispectral imaging sensor, and $\mathbf{E}_Y$ and $\mathbf{E}_Z$ are the errors. $\mathbf{X}$ is obtained by solving

$$\min_{\mathbf{X}} \frac{1}{2}\|\mathbf{Y} - \mathbf{XB}\|_F^2 + \frac{\alpha}{2}\|\mathbf{Z} - \mathbf{RX}\|_F^2 \tag{3}$$

where $\|\cdot\|_F$ represents the Frobenius norm, and $\alpha > 0$ balances the two fidelity terms.

HSI normally has a large correlation between bands [22], [25]. The N_B-dimensional spectral vectors of $\mathbf{X}$ usually are in a subspace of dimension much lower than N_B. Therefore

$$\mathbf{X} = \mathbf{AS} \tag{4}$$

where $\mathbf{A} \in \mathbb{R}^{N_B \times J}$ ($N_B \geq J$) represents the basis matrix whose J column vectors span the same subspace as the column vectors of $\mathbf{X}$, and $\mathbf{S} \in \mathbb{R}^{J \times N_W N_H}$ ($N_W N_H \geq J$) represents the coefficient matrix.

Equation (4) factors a matrix into two submatrices by either linear spectral unmixing (e.g., vertex component analysis [71]) or SVD [25]. To guide network design, only SVD is feasible here, since the order of the endmemebers extracted by linear spectral unmixing is interchangeable, which would disrupt subsequent predictions. Therefore, $\mathbf{A}$ is obtained by

$$[\mathbf{A}, \mathbf{\Sigma}, \mathbf{P}^T] = \mathrm{svds}(\mathbf{Y}, J) \tag{5}$$

where $\mathbf{A}$ and $\mathbf{P} \in \mathbb{R}^{N_w N_h \times J}$ are column orthogonal matrices, $\mathbf{\Sigma} \in \mathbb{R}^{J \times J}$ is a diagonal matrix containing the singular values, and $\mathrm{svds}(\cdot, J)$ is the truncated SVD function that keeps the J largest singular values, i.e., $\mathbf{Y} \approx \mathbf{A\Sigma P}^T$.

In deep learning-based fusion methods, the use of truncated SVD has three advantages. It can reduce the channels of input tensors, thus reducing computational load and storage requirements; it is a very common approach for denoising; and since the computation of SVD is performed on $\mathbf{Y}$ entirely, the estimates will normally bring extra information and global or nonlocal modules may not be needed [72], [73].

By incorporating (4) into the fusion process, (3) becomes

$$\min_{\mathbf{S}} \frac{1}{2}\|\mathbf{Y} - \mathbf{ASB}\|_F^2 + \frac{\alpha}{2}\|\mathbf{Z} - \mathbf{RAS}\|_F^2. \tag{6}$$

Problem (6) is still ill-posed, and cannot be solved for $\mathbf{X}$ and regularization terms need to be imposed on $\mathbf{S}$.

The coefficient $\mathbf{S}$ characterizes the spatial information of $\mathbf{X}$, and the column vectors of $\mathbf{S}$ correspond to those of $\mathbf{X}$ and so locate every pixel of $\mathbf{X}$. To locate pixels and capture more texture information, two regularization terms are used, and the final optimization problem can be written as

$$\min_{\mathbf{S}} \frac{1}{2}\|\mathbf{Y} - \mathbf{ASB}\|_F^2 + \frac{\alpha}{2}\|\mathbf{Z} - \mathbf{RAS}\|_F^2 + \lambda_l f_l(\mathbf{S}) + \lambda_t f_t(\mathbf{S}) \tag{7}$$

where $\lambda_l > 0$ and $\lambda_t > 0$ are the regularization parameters, and $f_l(\mathbf{S})$ and $f_t(\mathbf{S})$ (see Section III-C) represent the regularization functions that implement pixel localization and texture extraction.

B. Optimization Algorithm

The optimization problem (7) contains two fidelity terms and two regularization terms. ADMM can be used to solve this problem.[3] These terms can be divided into three groups based

[3]ADMM is used to solve (7) due to it is convenient to solve the problems with multiple regularization terms. Although other method, such as proximal gradient, can be used to solve (7), it has to evaluate a complex proximal operator consisting of two regularization terms, which increases the complexity of the problem.

on their functions: fidelity terms $f_y(\mathbf{S})$ and $f_z(\mathbf{S})$, regularization term $f_l(\mathbf{S})$ and regularization term $f_t(\mathbf{S})$, where $f_y(\mathbf{S}) = (1/2)\|\mathbf{Y} - \mathbf{ASB}\|_F^2$ and $f_z(\mathbf{S}) = (1/2)\|\mathbf{Z} - \mathbf{RAS}\|_F^2$. Therefore, with two variables $\mathbf{U}, \mathbf{\Pi} \in \mathbb{R}^{J \times N_W N_H}$, the optimization problem (7) can be rewritten as

$$\min_{\mathbf{S},\mathbf{U},\mathbf{V}} \; f_y(\mathbf{S}) + \alpha f_z(\mathbf{S}) + \lambda_l f_l(\mathbf{U}) + \lambda_t f_t(\mathbf{\Pi}) \quad \text{s.t. } \mathbf{U} = \mathbf{S}, \mathbf{\Pi} = \mathbf{S}. \tag{8}$$

The augmented Lagrangian function of (8) can be written as

$$\mathcal{L}(\mathbf{S}, \mathbf{U}, \mathbf{\Pi}, \mathbf{V}, \mathbf{\Lambda}) = f_y(\mathbf{S}) + \alpha f_z(\mathbf{S}) + \lambda_l f_l(\mathbf{U}) + \lambda_t f_t(\mathbf{\Pi}) + \frac{\mu}{2}\|\mathbf{S} - \mathbf{U} - \mathbf{V}\|_F^2 + \frac{\mu}{2}\|\mathbf{S} - \mathbf{\Pi} - \mathbf{\Lambda}\|_F^2 \tag{9}$$

where $\mathbf{V}, \mathbf{\Lambda} \in \mathbb{R}^{J \times N_W N_H}$ are auxiliary variables and $\mu > 0$ is the penalty parameter. The iteration procedures to solve (7) are given in detail below, and the entire process is shown in Algorithm 1.

1) Solving Subproblem $\mathbf{S}$*:* Optimizing $\mathcal{L}$ with respect to $\mathbf{S}$ can be written as

$$\min_{\mathbf{S}} f_y(\mathbf{S}) + \alpha f_z(\mathbf{S}) + \mu g_l(\mathbf{S}) + \mu g_t(\mathbf{S}) \tag{10}$$

where $g_l(\mathbf{S}) = (1/2)\|\mathbf{S} - \mathbf{U}^{(t-1)} - \mathbf{V}^{(t-1)}\|_F^2$, $g_t(\mathbf{S}) = (1/2)\|\mathbf{S} - \mathbf{\Pi}^{(t-1)} - \mathbf{\Lambda}^{(t-1)}\|_F^2$, and $t = 1, \ldots, T$ represents the tth ADMM iteration. Although problem (10) is convex and can be solved by a Sylvester equation [74], its solution is difficult to implement by a network. Equation (10) is solved by the gradient descent algorithm as

$$\mathbf{S}_k^{(t-1)} = \mathbf{S}_{k-1}^{(t-1)} - \eta\{\nabla_{\mathbf{S}} f_y(\mathbf{S}_{k-1}^{(t-1)}) + \alpha \nabla_{\mathbf{S}} f_z(\mathbf{S}_{k-1}^{(t-1)}) + \mu \nabla_{\mathbf{S}} g_l(\mathbf{S}_{k-1}^{(t-1)}) + \mu \nabla_{\mathbf{S}} g_t(\mathbf{S}_{k-1}^{(t-1)})\} \tag{11}$$

where

$$\nabla_{\mathbf{S}} f_y(\mathbf{S}_{k-1}^{(t-1)}) = \mathbf{A}^T(\mathbf{AS}_{k-1}^{(t-1)}\mathbf{B} - \mathbf{Y})\mathbf{B}^T \tag{12}$$

$$\nabla_{\mathbf{S}} f_z(\mathbf{S}_{k-1}^{(t-1)}) = (\mathbf{RA})^T(\mathbf{RAS}_{k-1}^{(t-1)} - \mathbf{Z}) \tag{13}$$

$$\nabla_{\mathbf{S}} g_l(\mathbf{S}_{k-1}^{(t-1)}) = \mathbf{S}_{k-1}^{(t-1)} - \mathbf{U}^{(t-1)} - \mathbf{V}^{(t-1)} \tag{14}$$

$$\nabla_{\mathbf{S}} g_t(\mathbf{S}_{k-1}^{(t-1)}) = \mathbf{S}_{k-1}^{(t-1)} - \mathbf{\Pi}^{(t-1)} - \mathbf{\Lambda}^{(t-1)} \tag{15}$$

$\eta > 0$ is the step, $k = 1, \ldots, K$ represents the kth subiteration, $\mathbf{S}_0^{(t-1)} = \mathbf{S}^{(t-1)}$ and $\mathbf{S}^{(t)} = \mathbf{S}_K^{(t-1)}$.

2) Solving Subproblem $\mathbf{U}$: Optimizing $\mathcal{L}$ with respect to $\mathbf{U}$ can be written as

$$\min_{\mathbf{U}} \frac{1}{2}\|\mathbf{U} - (\mathbf{S}^{(t)} - \mathbf{V}^{(t-1)})\|_F^2 + \frac{\lambda_l}{\mu} f_l(\mathbf{U}). \tag{16}$$

The solution of (16) can be achieved by a proximal operator $\text{prox}_{f_l}(\cdot, \cdot)$, i.e.,

$$\mathbf{U}^{(t)} = \text{prox}_{f_l}(\mathbf{S}^{(t)} - \mathbf{V}^{(t-1)}, \lambda_l/\mu). \tag{17}$$

3) Solving Subproblem $\mathbf{\Pi}$: Optimizing $\mathcal{L}$ with respect to $\mathbf{\Pi}$ can be written as

$$\min_{\mathbf{\Pi}} \frac{1}{2}\|\mathbf{\Pi} - (\mathbf{S}^{(t)} - \mathbf{\Lambda}^{(t-1)})\|_F^2 + \frac{\lambda_d}{\mu} f_d(\mathbf{\Pi}). \tag{18}$$

Algorithm 1: ADMM for Solving (7) Using the Lagrangian Formulation (9).

Input: LR-HSI $\mathbf{Y}$, HR-MSI $\mathbf{Z}$, $\mathbf{B}$, $\mathbf{R}$.
Calculate $\mathbf{A}$ by (5).
Initialize $\mathbf{S}^{(0)}$, $\mathbf{U}^{(0)}$, $\mathbf{V}^{(0)}$, $\mathbf{\Pi}^{(0)}$, $\mathbf{\Lambda}^{(0)}$ by zero matrix $\mathbf{0}$.
for $t = 1 : T$ **do**
  for $k = 1 : K$ **do**
    Update $\mathbf{S}_k^{(t)}$ by (11)–(15)
  end for
  Update $\mathbf{U}$ by (17)
  Update $\mathbf{\Pi}$ by (19)
  Update $\mathbf{V}$ by (20)
  Update $\mathbf{\Lambda}$ by (21)
end for
Output: The coefficient matrix $\mathbf{S}$.

Similar to (16), the solution of (18) is achieved by a proximal operator $\text{prox}_{f_d}(\cdot, \cdot)$, i.e.,

$$\mathbf{\Pi}^{(t)} = \text{prox}_{f_d}(\mathbf{S}^{(t)} - \mathbf{\Lambda}^{(t-1)}, \lambda_d/\mu). \tag{19}$$

4) Updating Multipliers: The multipliers associated with $\mathcal{L}$ are updated as

$$\mathbf{V}^{(t)} = \mathbf{V}^{(t-1)} - (\mathbf{S}^{(t)} - \mathbf{U}^{(t)}) \tag{20}$$

$$\mathbf{\Lambda}^{(t)} = \mathbf{\Lambda}^{(t-1)} - (\mathbf{S}^{(t)} - \mathbf{\Pi}^{(t)}). \tag{21}$$

C. Deep Fusion Net

Based on Algorithm 1, we build a deep HS/MS image fusion network by unfolding all steps of the algorithm as network layers. The proposed network is mainly a structure of T stages, corresponding to T ADMM iterations (see Fig. 1). Each stage contains three modules (the $\mathbf{S}$, $\mathbf{U}$, and $\mathbf{\Pi}$ modules) and two computation units (the $\mathbf{V}$ and $\mathbf{\Lambda}$ units), which represent the five procedures of one ADMM iteration. It takes $\mathbf{Y}$, $\mathbf{Z}$, $\mathbf{A}$, and the outputs of previous stage, as inputs, and outputs five updated variables to be the inputs of next stage. The final $\mathbf{X}$ is obtained by convolving and then averaging the fused images $\mathbf{X}^{(t)}$ produced in all stages. This process is called *aggregation fusion*. The techniques adopted for the proposed approach will be described in more detail.

1) $\mathbf{S}$ *Module:* Fig. 2 shows the structure of the $\mathbf{S}$ module. The module is a structure of K stages, corresponding to K iterations of the gradient descent algorithm. Equation (12) can be expanded as

$$\nabla_{\mathbf{S}} f_y(\mathbf{S}_{k-1}^{(t-1)}) = (\mathbf{S}_{k-1}^{(t-1)}\mathbf{B} - \mathbf{A}^T\mathbf{Y})\mathbf{B}^T \tag{22}$$

where the identity matrix $\mathbf{A}^T\mathbf{A}$ is omitted. In (22), $\mathbf{B}$ is performed using a 2-D strided convolution followed by a Leaky ReLU, since $\mathbf{B}$ represents the spatial blurring and downsampling operator. Similarly, $\mathbf{B}^T$ is performed using a 2-D strided deconvolution. To fit the resolution ratio, the stride is fixed to r and the kernel size is set as $1.5\,r \times 1.5\,r$. To reduce parameters and keep the flexibility of model, the parameters of the pipeline

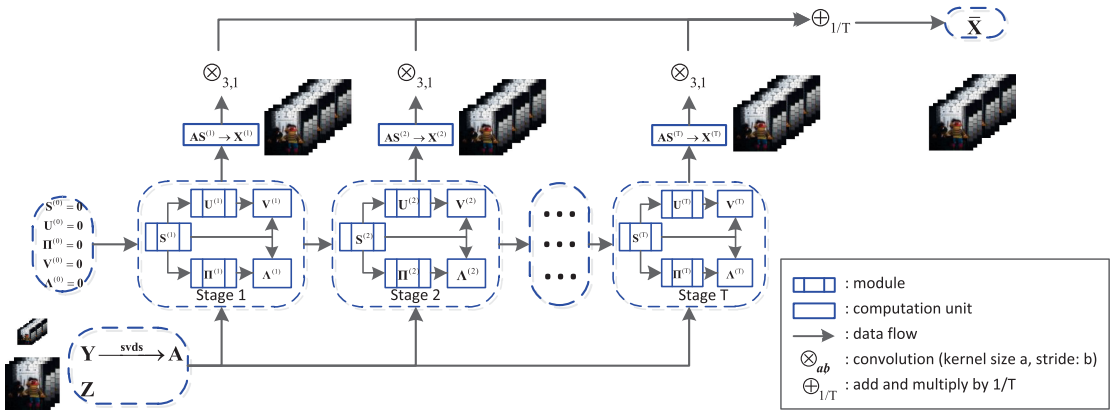


Fig. 1. Overall structure of the proposed network.

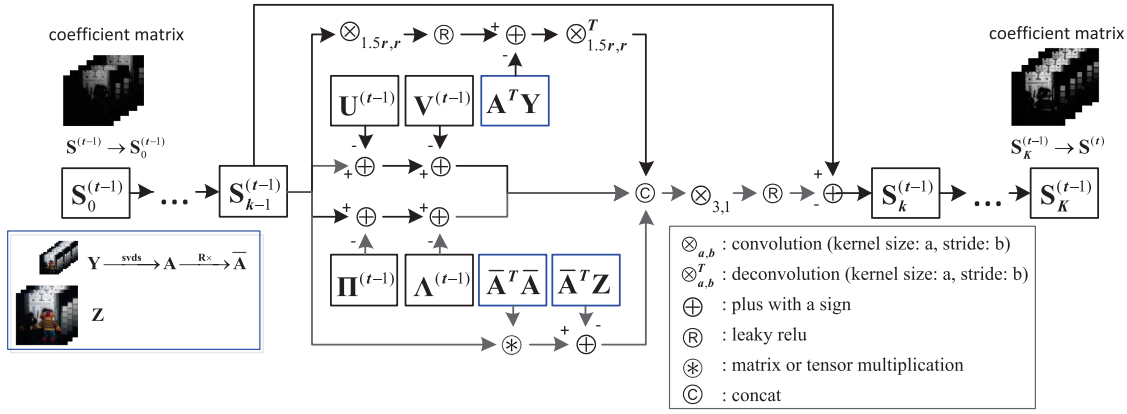


Fig. 2. Structure of the coefficient matrix $\mathbf{S}$ module.

(22) are only shared in ADMM iterations. Equation (13) can be rewritten as

$$\nabla_{\mathbf{S}} f_z(\mathbf{S}_{k-1}^{(t-1)}) = \bar{\mathbf{A}}^T \bar{\mathbf{A}} \mathbf{S}_{k-1}^{(t-1)} - \bar{\mathbf{A}}^T \mathbf{Z} \tag{23}$$

where $\bar{\mathbf{A}} = \mathbf{R}\mathbf{A}$. In (23), $\mathbf{R}$ is a parameter matrix learned by the network, which is variable in different ADMM iterations and unchanged in the gradient descent process. Equations (14) and (15) can be implemented directly by the network.

In (11), a linear combination of four gradient functions is calculated and followed by gradient descent. The linear combination is derived from the fixed format of (3). To break the fixed format and provide more flexibility, the linear combination, and step η, are implemented by concatenating the four gradient functions and performing a 3×3 2-D convolution and a Leaky ReLU. Then, one gradient descent can be performed by a subtraction operation.

2) $\mathbf{U}$ and $\mathbf{\Pi}$ Modules: Equations (17) and (19) are two proximal operators. Equation (17) is derived from the regularization term $f_l(\mathbf{S})$ for pixel localization. A U-net architecture [75] is used to learn this operator, with residual connections [76] introduced to improve performance. Equation (19) is derived from the regularization term $f_t(\mathbf{S})$ for texture extraction, which can be treated as the localization of subpixels, and thus a reversed U-net architecture is used to learn this operator. Fig. 3 shows the structures of the $\mathbf{U}$ and $\mathbf{\Pi}$ modules that appear as twin U-nets.

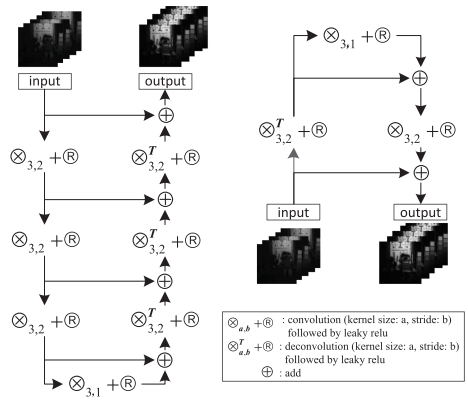


Fig. 3. Structures of the $\mathbf{U}$ and $\mathbf{\Pi}$ modules. Left: U-net $\mathbf{U}$. Right: reversed U-net $\mathbf{\Pi}$.

To simplify the model, the number of feature channels is kept unchanged in both modules. For the number of 2-times sampling operators, it is set as $\log_2 r$ for the $\mathbf{U}$ module (Fig. 3 shows the structure when $r = 8$) and it is fixed to 1 for the $\mathbf{\Pi}$ module. The computation units of the auxiliary variables of $\mathbf{U}$ and $\mathbf{\Pi}$, $\mathbf{V}$, and $\mathbf{\Lambda}$, can be implemented directly.

Both $\mathbf{U}$ and $\mathbf{\Pi}$ modules are implemented for spatial injection, but for different purposes. U-net is a basic architecture for

many image segmentation tasks, which enables precise localization [75]. As shown in [77], U-net architecture can also be utilized to capture the main structure of images. Implemented by a U-net, $\mathbf{U}$ module is designed to capture the spatial information ranging from the pixel to region level. Image texture can be seen as a subpixel-level structure, which is also important for spatial enhancement. Inspired by subpixel mapping [78], texture extraction can be treated as the localization of subpixels. Reversed U-net that upsamples the image gradually can achieve this purpose. Implemented by a reversed U-net, $\mathbf{\Pi}$ module is designed to capture the subpixel-level information and can be seen as a complement to $\mathbf{U}$ module.

3) Aggregation Fusion: After the pipelines of the ADMM iterations, the desired $\mathbf{X}$ can be produced by a straightforward use of the final $\mathbf{S}^{(T)}$, along with $\mathbf{A}$. However, the subspace-based fusion model (7) mainly focuses on the injection of spatial information rather than spectral information. Although the basis matrix $\mathbf{A}$ describes the spectral information of HR-HSI, it is insufficient to obtain the desired result, since there may be the loss of critical information or the inconsistency between spectral and spatial information after network inference. A further spectral-spatial fusion must be performed on the entire image. As image averaging can reduce the zero-mean noise in digital image processing if multiple images are taken independently, this will be suitable for the fused images produced at the different stages. The output of the proposed network can be written as

$$\bar{\mathbf{X}} = \frac{1}{T}\sum_{t=1}^{T} \otimes_{3,1}(\mathbf{X}^{(t)}) \tag{24}$$

where $\mathbf{X}^{(t)} = \mathbf{A}\mathbf{S}^{(t)}$ and $\otimes_{3,1}$ is a 2-D convolution operator with a kernel size of 3×3 and stride of 1. To measure the difference between the output $\bar{\mathbf{X}}$ and the target $\mathbf{X}$, the l_1-norm is used because it is more robust to outliers than the l_2-norm. Then, the final objective function can be written as

$$\begin{aligned} &\min_{\Theta} \ \|\mathbf{X} - \bar{\mathbf{X}}\|_1 \\ &\text{s.t. } \bar{\mathbf{X}} = \mathrm{SpfNet}_{\Theta}(\mathbf{Y}, \mathbf{Z}, \mathbf{A}) \end{aligned} \tag{25}$$

where Θ-parameterized $\mathrm{SpfNet}_{\Theta}(\cdot)$ represents the proposed network. *Aggregation fusion* is also required in the test phase. Existing deep learning-based HS/MS image fusion approaches input the entire observation images to produce the fused image in the test phase, while small patches are used in the training and validation phases. This simplifies the calculation, but the model of the training phase and the accuracy of the validation phase will not match those of the test phase. Moreover, it is inappropriate to fuse the images entirely due to the significant spectral differences of distant locations, and it is also not suitable for online processing. In the proposed approach, the test images are cut (with a stride of L) into several overlapping patches of the same size as those used for training,[4] and the desired HR-HSI is obtained by tiling and summation over all generated patches. There are two advantages to this test strategy: the training and test phases are consistent so the accuracy of the validation set can more precisely reflect that of the testset, and, with the assistance of SVD, the generated pixels that belong to different patches will have different information, so the performance of the network can be improved by the aggregation of patches.

IV. Experimental Results and Analysis

In this section, experiments on a synthetic dataset are conducted to verify the mechanism of SpfNet quantitatively, and then its performance is evaluated on synthetic and real datasets. All datasets were scaled to the range [0, 1]. The quality of the fused images in the synthetic datasets was assessed with five quantitative indices, root-mean-squared error (rmse), peak signal-noise-ratio (PSNR), spectral angle mapper (SAM),[5] structural similarity index (SSIM), and relative dimensionless global error in synthesis (ERGAS) [9], [10].

A. Synthetic Datasets

Three real-life HSI datasets, CAVE [79], Harvard [80], and Pavia Center (PaviaC)[6] were manipulated to use as synthetic reference images for the simulation experiments.

1) The CAVE dataset consists of 32 indoor HSIs of 512×512 pixels, which contain 31 spectral bands acquired at wavelength intervals of 10 nm in the range of 0.4–0.7 μm. The first 20 HSIs were used for training, and the remainder for testing.
2) The Harvard dataset contains 50 HSIs of indoor and outdoor scenes under daylight illumination. Each HSI has 1392×1040 pixels and 31 spectral bands. The spectral range covers 0.42–0.72 μm with a wavelength interval 10 nm. The first 30 HSIs were used for training, and the remainder for testing.
3) The PaviaC dataset is an urban image acquired by the reflective optics system imaging spectrometer (ROSIS), with a spectral range of 0.43 to 0.86 μm. The ROSIS sensor gives 115 spectral bands and 103 remained after removal of noisy bands. The size of the HSI is 1096×1096 pixels, but the central area contains no information and has to be discarded, resulting in two subimages of 1096×223 and 1096×492 pixels. The bottom-left 512×216-pixel part of the image was selected as the test image, and the remaining parts were used for training.

For each reference image, two observation images, LR-HSI and HR-MSI, were generated according to Wald's protocol [81]. To generate the LR-HSI, the reference image is blurred and down-sampled by a factor of 8 ($r = 8$) in each direction. A Gaussian blur of 15×15 pixels, with a mean of 0 and a standard deviation of 3.40, was applied to each band of the reference image. To generate the HR-MSI, the band number N_B of the reference image $\mathbf{X} \in \mathbb{R}^{N_B \times N_W N_H}$ is reduced to N_b by left multiplying a spectral response function $\mathbf{R} \in \mathbb{R}^{N_b \times N_B}$. For the

[4]The patch sizes are the same in the training and test phases, but the strides are not necessarily the same. In the training phase the stride controls the number of training samples, while in the test phase the stride affects the quality as shown in Section IV-C1.

[5]We compute the SAM between two pixels in degree.

[6][Online]. Available: http://www.ehu.eus/ccwintco/index.php/Hyperspectral_Remote_Sensing_Scenes#Pavia_Centre_scene

first two datasets, $\mathbf{R}$ was derived from the spectral response of a Nikon D700 camera.[7] For the last dataset, $\mathbf{R}$ was derived from the spectral response of the IKONOS satellite, according to the spectral response profiles of the RGB and NIR bands. Specifically, the HR-MSI has three bands in the CAVE and Harvard datasets and four bands in the PaviaC dataset. The above simulation process of generating the LR-HSI and HR-MSI cannot match their degradation process well and there may be ubiquitous noise or error, due to the complexity of real scenarios. To better simulate the degradation process, moderate Gaussian noise was added to the LR-HSI (SNR $=30$ dB) and the HR-MSI (SNR $=40$ dB).

To prepare samples for training, 64×64 pixel overlapping patches from the reference images were extracted as the desired HR-HSIs. The stride sizes of extracted patches were 16 for the CAVE dataset, 48 for the Harvard dataset, and 8 for the PaviaC dataset. Then the utilized LR-HSIs and HR-MSIs are of sizes 8×8 and 64×64 pixels, respectively. About 20% of these extracted patches were used for validation.

B. Comparison Methods and Implementation Details

Seven approaches, which can be divided into model- and deep learning-based approaches, were compared to evaluate the performance of SpfNet.[8] The model-based approaches are coupled spectral unmixing (CSU[9]) [23], HySure[10] [25], NPTSR[11] [19], and CNNFUS[12] [30]. The deep learning-based approaches are PNN[13] [39], MHFnet[14] [61], and DBIN[15] [69], and also SpfNet. The free parameters of the model-based approaches were tuned to be optimal with the test datasets and the default parameters were used for the deep learning-based approaches. Specifically, leaving the default parameters unchanged, in CSU the number of endmembers was set as 30 for all datasets; in HySure the dimension of subspace was set as 10 for all datasets; in NPTSR the sizes of patch and step were fixed to 32×32 and 16, λ and β were set as 10^{-2} and 10^{2} for the CAVE dataset, 10^{-3} and 10^{2} for the Harvard dataset, and 10^{2} and 10^{4} for the PaviaC dataset; in CNNFUS the parameters were set as $\lambda = 10^{-4}$, $T = 25$ for all datasets and as $L = 10, 5, 5$ for the CAVE, Harvard and PaviaC datasets, respectively. For SpfNet, unless otherwise specified, the default parameters were used, that is, the number of ADMM iterations, T, was fixed to 5, the number of gradient descent iterations, K, was fixed to 3, and the column number, J, of the basis matrix $\mathbf{A}$ was set as $\min\{31, N_B\}$. The model-based approaches were performed using MATLAB, and the deep networks were implemented by the TensorFlow framework with Python. All deep learning-based approaches used the same training and valid sets, and an Adam optimizer was used to train the networks with a batch size of 32. For MHFnet, it was trained for 100 epochs with learning rate being 10^{-4}. For PNN, DBIN and SpfNet, they were trained for 200 epochs, and the learning rate was initialized at 10^{-3} and gradually decayed to 5×10^{-4}, 10^{-4} and 5×10^{-5}. Fig. 4 shows the training and valid losses as a function of epochs for the proposed SpfNet. As the number of epochs increases, the two losses decrease consistently.

TABLE I
QUALITY MEASURES USING DIFFERENT SUBSPACE DIMENSIONS

Dimension	RMSE	PSNR	SAM	SSIM	ERGAS
Best Values	0	$+\infty$	0	1	0
CAVE					
5	0.00571	45.503	3.899	0.991	0.688
10	0.00573	45.460	3.786	0.991	0.697
15	0.00526	46.237	3.574	0.992	0.637
20	0.00512	46.457	3.471	0.992	0.621
25	0.00499	46.709	3.367	0.993	0.603
30	0.00497	46.749	3.309	0.993	0.602
31	**0.00491**	**46.839**	**3.281**	**0.993**	**0.596**
PaviaC					
10	0.00736	42.660	2.777	0.985	0.769
20	0.00732	42.713	2.758	0.985	0.766
30	**0.00724**	**42.804**	**2.743**	0.985	0.763
40	0.00736	42.657	2.767	0.985	0.777
50	0.00727	42.773	2.746	**0.985**	**0.762**
64	0.00733	42.698	2.755	0.985	0.767

C. Parameter Analysis and Ablation Study

In this section, we will use a set of experiments to show the influence of the key parameters, the patch size, the test stride L, the subspace dimension J and the ADMM stage T. We also present the efficiency of SpfNet via an ablation study. The experiments keep the default parameters and structures, except for the parameter or structure being assessed.

1) Influence of the Patch Size and Test Stride L: The use of moderate-sized test patches is a part of the *aggregation fusion* technique (Section III-C3). This experiment showed how the patch size and test stride affect the performance of SpfNet on the CAVE dataset. The quality measures PSNR and SAM are illustrated in Fig. 5. It can be seen that moderate-sized 64×64 patch is better than the others. For all patch sizes, performance was best at a stride of 8, and became steadily worse as the stride increased or decreased. To balance accuracy and computational overhead, the patch size was fixed to 64×64, and the test stride L was set as 16 for the CAVE dataset and as 16 and 4 for the Harvard and PaviaC datasets, respectively.

2) Influence of J: The subspace dimension J determines the number of input channels for all modules, giving the scale of the network. LR-HSI patches of 8×8 were used in the experiments. Thus, the dimension cannot be greater than 64, regardless of the number of spectral bands. The CAVE and Harvard datasets have 31 spectral bands, and thus, J cannot be greater than 31. The PaviaC dataset uses 103 spectral bands, and thus, J cannot be greater than 64. Table I shows the results as a function of J for the CAVE and PaviaC datasets. In this table and the following, the best values are marked in bold. Performance on the CAVE dataset, as judged by the quality

[7] [Online]. Available: https://maxmax.com/spectral_response.htm
[8] The code will be available on https://github.com/liuofficial
[9] [Online]. Available: https://github.com/lanha/SupResPALM
[10] [Online]. Available: https://github.com/alfaiate/HySure
[11] The code is provided by Dr. Xu
[12] [Online]. Available: https://github.com/renweidian/CNN-FUS
[13] [Online]. Available: https://github.com/sergiovitale/pansharpening-cnn-python-version
[14] [Online]. Available: https://github.com/XieQi2015/MHF-net
[15] [Online]. Available: https://github.com/wwhappylife/Deep-Blind-Hyperspectral-Image-Fusion

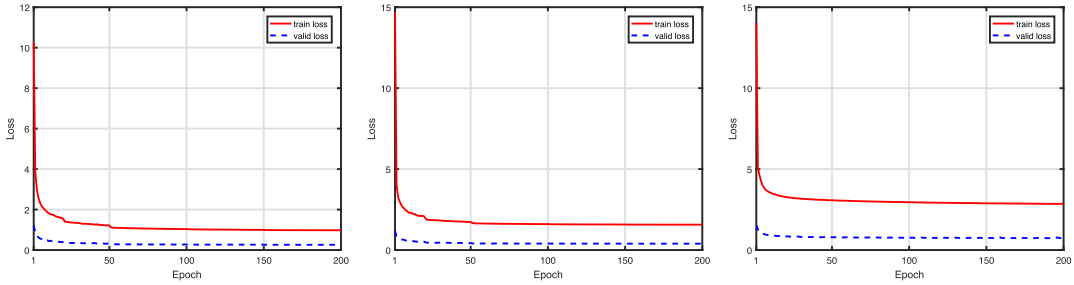


Fig. 4. Training and valid losses as a function of epochs. (a) CAVE dataset. (b) Harvard dataset. (c) PaviaC dataset.

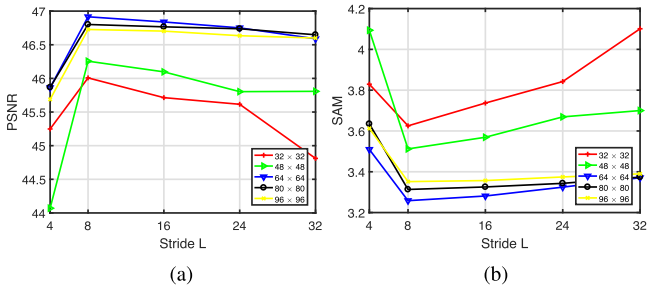


Fig. 5. Quality measures as a function of the stride L using different patch sizes. (a) PSNR. (b) SAM.

TABLE II
QUALITY MEASURES FOR THE CAVE DATASET USING DIFFERENT STAGES

Stage	RMSE	PSNR	SAM	SSIM	ERGAS
Best Values	0	$+\infty$	0	1	0
1	0.00599	45.045	4.017	0.990	0.730
3	0.00516	46.388	3.525	0.992	0.626
5	**0.00491**	**46.839**	**3.281**	**0.993**	**0.596**
7	0.00497	46.758	3.333	0.993	0.601
9	0.00509	46.565	3.354	0.993	0.615

measures, increases as the dimension increases and is best at full rank; while on the PaviaC dataset, it is best at 30. Without loss of generality, J was set as $\min\{31, N_B\}$ for all datasets.

3) Influence of T: The ADMM stage T controls the depth of the network. For the iterative algorithm, a large T is beneficial. This experiment showed how the stage affects the performance of SpfNet on the CAVE dataset. Performance, on all quality measures, increases as the stage increases to five and then slightly decreases (see Table II). In the experiments, the ADMM stage T was fixed as five for all datasets.

4) Ablation Study: Subspace representation, coefficient matrix $\mathbf{S}$ module, U-net $\mathbf{U}$ module, reversed U-net $\mathbf{\Pi}$ module and aggregation fusion are parts of the design of the network in SpfNet. The contribution of each technique to the network was assessed on the quality measures by removing it from the network. Specifically, seven different structures are investigated. "SVD" denotes the proposed SpfNet without performing SVD on patches. "$\mathbf{U}$," "$\mathbf{\Pi}$," and "$\mathbf{U}+\mathbf{\Pi}$" denote the proposed SpfNet without using the corresponding modules. "$\mathbf{S}$-lin" refers to $\mathbf{S}$ module using a linear combination of the four gradient functions in (11). "AF-average" refers to the HR-HSI $\mathbf{X}$ directly computed using the coefficient matrix $\mathbf{S}^{(T)}$ and the basis matrix $\mathbf{A}$ rather than image averaging. "AF-patch" is the way in which the test images are fused entirely rather than patch by patch. The complete SpfNet performs better than all models with a component removed indicating that all techniques contribute positively to the final result (see Table III).

TABLE III
ABLATION STUDY USING DIFFERENT STRUCTURES WHEN APPLIED TO THE CAVE DATASET

Structure	RMSE	PSNR	SAM	SSIM	ERGAS
Best Values	0	$+\infty$	0	1	0
SVD	0.00512	46.455	3.462	0.992	0.623
$\mathbf{U}$	0.00501	46.661	3.405	0.993	0.609
$\mathbf{\Pi}$	0.00505	46.624	3.410	0.993	0.611
$\mathbf{U}+\mathbf{\Pi}$	0.00531	46.145	3.727	0.991	0.643
$\mathbf{S}$-lin	0.00540	46.016	3.645	0.992	0.657
AF-average	0.00565	45.616	4.007	0.990	0.682
AF-patch	0.00643	44.342	4.637	0.988	0.780
SpfNet	**0.00491**	**46.839**	**3.281**	**0.993**	**0.596**

D. Results of Experiments on Synthetic Datasets

The seven approaches mentioned in Section IV-B were compared quantitatively and visually with SpfNet, based on the quality measures, in four groups. The first contains the four model-based methods, CSU, HySure, NPTSR, and CNNFUS using the exact spatial blur and spectral response ($\mathbf{B}$ and $\mathbf{R}$) while the second group is the same methods (indicated by a suffix "-B," for a blind group) where $\mathbf{B}$ and $\mathbf{R}$ were estimated as in [25]. The third group contains the three deep learning-based methods, PNN, MHFnet, and DBIN, indicated with a suffix "-S" to show that small patches were used in the test phase, as in SpfNet. The fourth group is the four deep learning-based methods, PNN, MHFnet, DBIN, and SpfNet, under their standard test conditions.

On the CAVE dataset, SpfNet performed the best followed by DBIN and MHFnet (see Table IV). Model-based methods were greatly influenced by $\mathbf{B}$ and $\mathbf{R}$. The use of small test patches gave worse results for PNN, MHFnet, and DBIN. For blind fusion, the deep-learning methods performed better than the model-based methods. Fig. 6 illustrates the fusion results for the image 'thread_spools' as RGB images, according to the spectral response of a Nikon D700 camera for the model based methods with estimated $\mathbf{B}$ and $\mathbf{R}$ and the standard deep learning

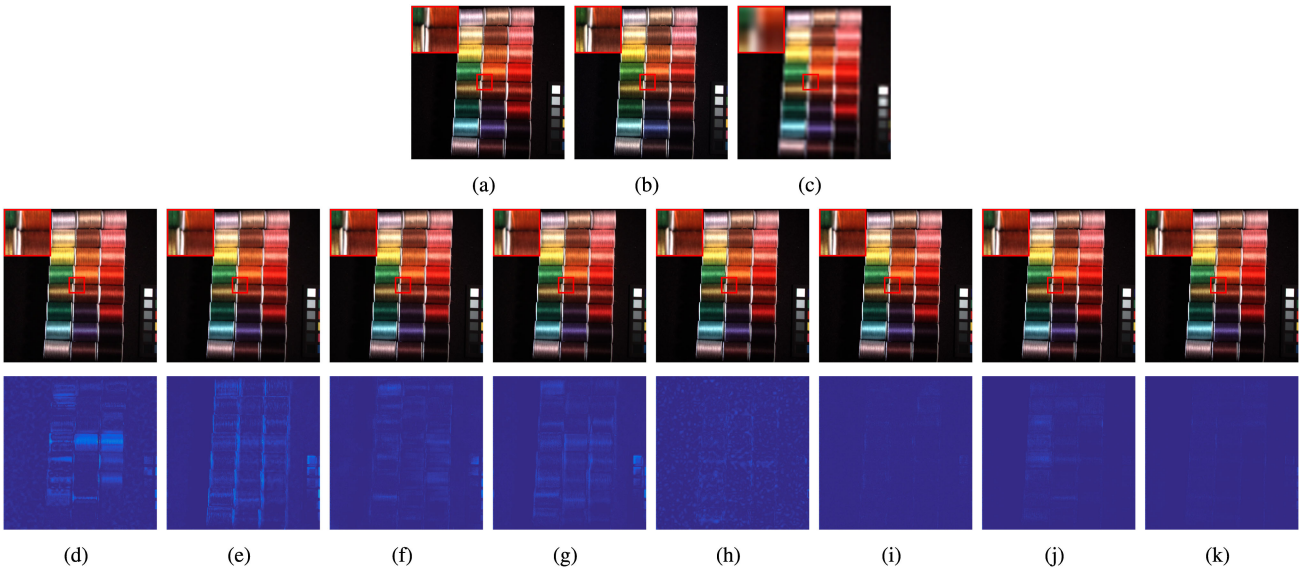


Fig. 6. RGB images (with a meaningful region marked and zoomed in 3 times for easy observation) and error maps (at band 15) of HS/MS image fusion results when applied to the CAVE dataset. (a) Reference image. (b) HR-MSI. (c) LR-HSI. (d) CSU-B. (e) HySure-B. (f) NPTSR-B. (g) CNNFUS-B. (h) PNN. (i) MHFnet. (j) DBIN. (k) SpfNet.

TABLE IV
QUALITY MEASURES FOR THE CAVE DATASET USING DIFFERENT METHODS

Method	RMSE	PSNR	SAM	SSIM	ERGAS
Best Values	0	$+\infty$	0	1	0
CSU	0.00907	41.303	6.480	0.979	1.111
HySure	0.00711	43.496	4.864	0.986	0.864
NPTSR	0.00688	43.827	4.969	0.985	0.830
CNNFUS	0.00733	43.077	4.987	0.986	0.914
CSU-B	0.02025	36.714	8.849	0.948	2.184
HySure-B	0.01729	37.510	7.281	0.950	1.861
NPTSR-B	0.01247	39.862	7.348	0.969	1.448
CNNFUS-B	0.01411	38.707	7.650	0.966	1.618
PNN-S	0.00852	41.973	8.491	0.968	1.006
MHFnet-S	0.00906	41.688	6.824	0.964	1.074
DBIN-S	0.00666	44.486	3.589	0.992	0.804
PNN	0.00851	41.986	8.494	0.968	1.005
MHFnet	0.00609	44.900	5.503	0.986	0.735
DBIN	0.00660	44.582	3.580	0.992	0.796
SpfNet	**0.00491**	**46.839**	**3.281**	**0.993**	**0.596**

TABLE V
QUALITY MEASURES FOR THE HARVARD DATASET USING DIFFERENT METHODS

Method	RMSE	PSNR	SAM	SSIM	ERGAS
Best Values	0	$+\infty$	0	1	0
CSU	0.00901	42.608	4.349	0.977	1.409
HySure	0.00840	43.194	3.880	0.977	1.369
NPTSR	0.00722	44.993	3.545	0.981	1.142
CNNFUS	0.00780	44.263	3.447	0.981	1.193
CSU-B	0.01140	40.785	4.625	0.971	1.592
HySure-B	0.01454	38.800	4.685	0.943	2.738
NPTSR-B	0.00858	43.233	3.759	0.977	1.618
CNNFUS-B	0.01148	41.073	3.881	0.968	1.709
PNN-S	0.00920	42.193	5.625	0.970	1.560
MHFnet-S	0.00965	41.491	5.087	0.964	1.958
DBIN-S	0.00719	45.038	3.118	0.983	1.066
PNN	0.00919	42.204	5.633	0.970	1.557
MHFnet	0.00727	44.895	3.418	0.982	1.075
DBIN	0.00719	45.043	3.118	0.983	1.067
SpfNet	**0.00693**	**45.479**	**3.018**	**0.983**	**1.029**

methods. The reference image, the HR-MSI and the LR-HSI are also presented. SpfNet performs well, and spectral distortion is not evident for the methods tested. Fig. 9(a) shows PSNR as a function of the spectral band for the methods used in Fig. 6. For almost all bands, SpfNet performs best followed by MHFnet. The SAMs between the reference image and the fusion results for each pixel using these methods, are shown in Fig. 10(a), with the pixels sorted by ascending error. SpfNet outperforms the other methods at the pixel level.

For the Harvard dataset, for all quality measures, SpfNet performed the best followed by DBIN (see Table V). The model-based methods performed differently between the blind and nonblind groups. MHFnet was sensitive to the size of test images while PNN and DBIN were not. Fig. 7 shows the original 'imge6' images and the fusion results of the eight blind methods as RGB images. There is no obvious spectral distortion for all methods. Fig. 9(b) gives PSNR as a function of the spectral band for these blind methods. SpfNet, DBIN, and MHFnet achieved high results in most bands. Fig. 10(b) gives the SAMs for each pixel between the reference image and the fusion. SpfNet obtains consistently good results.

On the PaviaC dataset, under the five quality measures, SpfNet, MHFnet-S, and DBIN-S achieved high results with no significant differences among them (see Table VI). The performance of the model-based methods degrades substantially in the blind scenario. When using small test patches, the performance of MHFnet and DBIN is improved, while there is no significant difference for PNN. RGB images of the reference HR-HSI, the HR-MSI, the LR-HSI, and the fusion results of the eight blind methods, according to the spectral response of the IKONOS

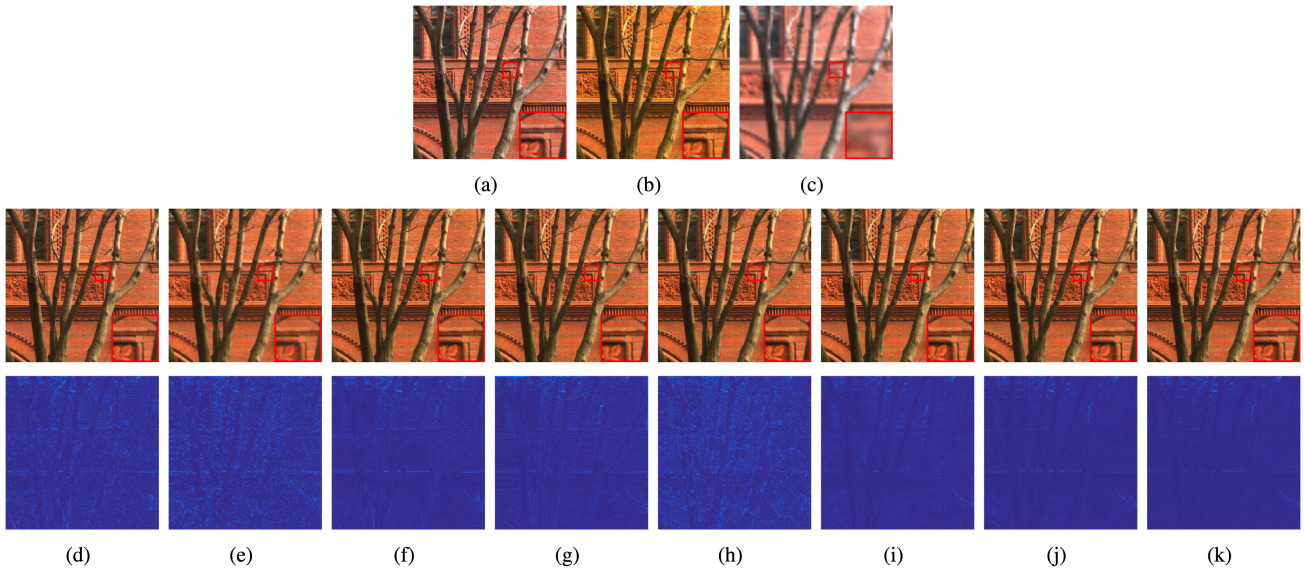


Fig. 7. RGB images (with a meaningful region marked and zoomed in 3 times for easy observation) and error maps (at band 15) of HS/MS image fusion results when applied to the Harvard dataset. (a) Reference image. (b) HR-MSI. (c) LR-HSI. (d) CSU-B. (e) HySure-B. (f) NPTSR-B. (g) CNNFUS-B. (h) PNN. (i) MHFnet. (j) DBIN. (k) SpfNet.

TABLE VI
QUALITY MEASURES FOR THE PAVIAC DATASET USING DIFFERENT METHODS

Method	RMSE	PSNR	SAM	SSIM	ERGAS
Best Values	0	$+\infty$	0	1	0
CSU	0.01339	37.465	3.318	0.977	1.339
HySure	0.01671	35.541	5.785	0.956	1.377
NPTSR	0.00789	42.059	3.020	0.984	0.806
CNNFUS	0.00987	40.112	3.414	0.981	0.989
CSU-B	0.02601	31.697	4.260	0.947	2.273
HySure-B	0.04385	27.161	6.894	0.741	3.897
NPTSR-B	0.01474	36.630	5.039	0.969	1.473
CNNFUS-B	0.02261	32.916	4.598	0.938	2.138
PNN-S	0.01050	39.575	4.054	0.977	1.061
MHFnet-S	0.00711	42.961	2.790	0.985	0.768
DBIN-S	**0.00709**	**42.986**	2.773	0.985	0.762
PNN	0.01073	39.392	4.143	0.977	1.085
MHFnet	0.00756	42.435	3.003	0.983	0.823
DBIN	0.00791	42.042	3.120	0.982	0.865
SpfNet	0.00716	42.905	**2.743**	**0.985**	**0.752**

TABLE VII
TRAINING AND TEST TIMES OF EACH METHOD

	Training (hours)			Test (seconds)		
	CAVE	Harvard	PaviaC	CAVE	Harvard	PaviaC
CSU	–	–	–	189.5	828.1	88.8
HySure	–	–	–	118.3	567.3	44.3
NPTSR	–	–	–	218.2	1518.9	661.7
CNNFUS	–	–	–	10.2	29.4	3.4
PNN	**2.7**	**2.8**	4.6	**0.3**	**0.7**	**2.1**
MHFnet	14.4	15.1	16.9	0.6	1.8	5.9
DBIN	12.5	13.1	9.3	0.6	2.5	2.8
SpfNet	6.2	6.5	**3.9**	3.4	20.5	25.6

TABLE VIII
NUMBER OF TRAINABLE PARAMETERS (IN MILLIONS)

	PNN	MHFnet	DBIN	SpfNet
CAVE	**0.2**	2.4	3.0	1.8
Harvard	**0.2**	2.4	3.0	1.8
PaviaC	**0.5**	18.9	7.9	2.3

satellite, are given in Fig. 8. No method exhibits an obvious spectral distortion. PSNR and SAM, as functions of the spectral band and by pixel sorted on error, are shown in Figs. 9(c) and 10(c), respectively. SpfNet, DBIN, and MHFnet consistently outperform the other methods.

E. Computational Efficiency

The experiments in Section IV were carried out using a desktop computer with an Intel Core i9-7900X CPU (3.3 GHz, 10 cores), a GeForce GTX 2080Ti GPU, and 64-GB memory. Table VII summarizes the test times of the compared methods mentioned in Section IV-B and the training times of the related deep learning-based methods, and the number of trainable parameters for each deep learning-based method is reported in Table VIII. PNN is the fastest method in most cases, and CNNFUS is the fastest one among the model-based methods. The proposed SpfNet is slightly slower than the other deep learning-based methods, since it divides the test images into overlapping patches for fusion.

F. Results of Experiments on the Real Dataset

The World View-2 (WV2) dataset[16] was used to evaluate SpfNet on real data. This dataset consists of a LR-HSI of 419 $\times$ 658 $\times$ 8 and a high-spatial-resolution RGB (HR-RGB) image of 1676 $\times$ 2632 $\times$ 3, with a resolution ratio of 4. Take the HR-RGB image as a reference, the bottom-right 512 $\times$ 512 pixel image was taken as the test image and the remaining

[16][Online]. Available: https://www.l3harrisgeospatial.com/Data-Imagery/Satellite-Imagery/High-Resolution/WorldView-2

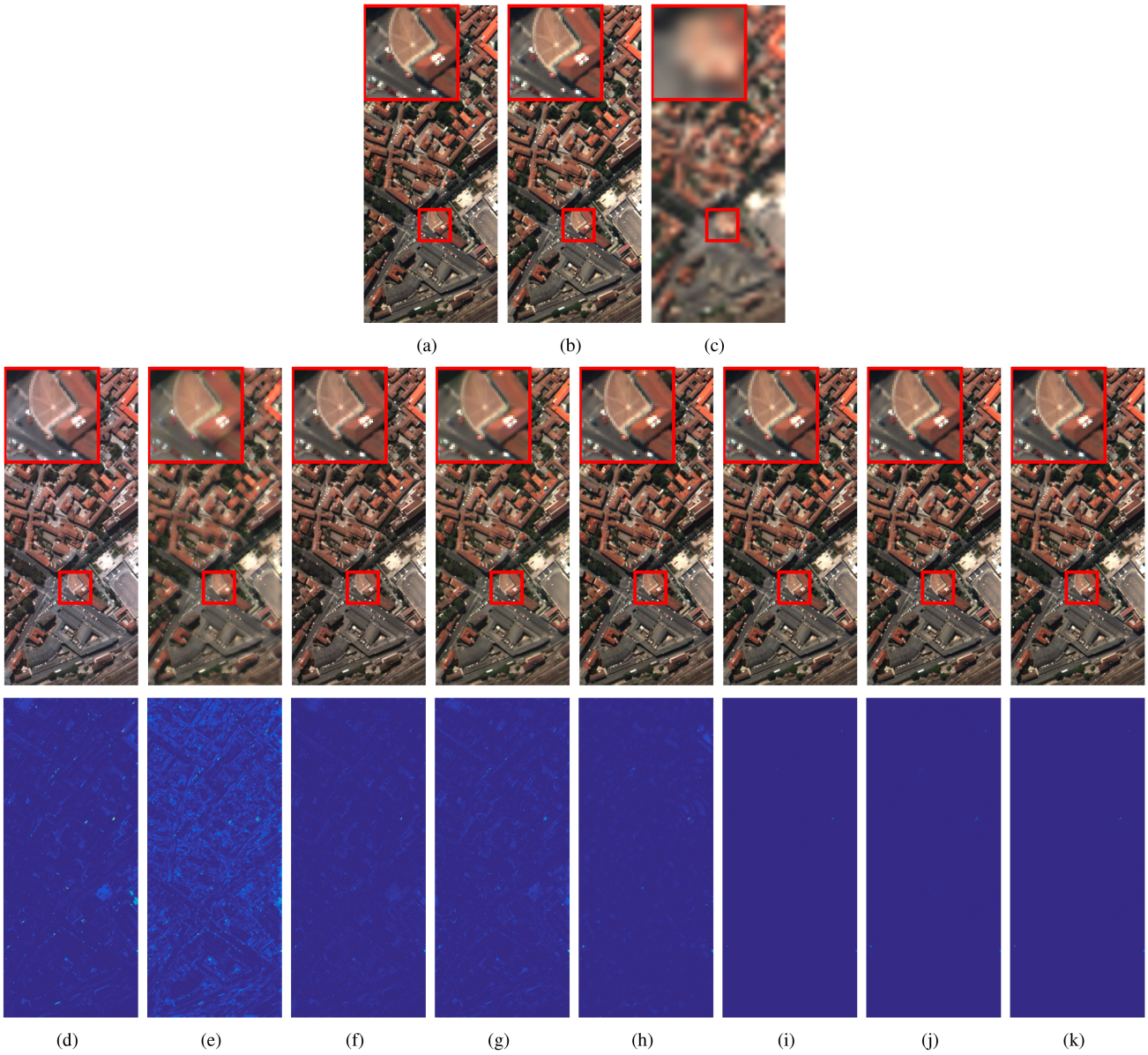


Fig. 8. RGB images (with a meaningful region marked and zoomed in 3 times for easy observation) and error maps (at band 30) of HS/MS image fusion results when applied to the PaviaC dataset. (a) Reference image. (b) HR-MSI. (c) LR-HSI. (d) CSU-B. (e) HySure-B. (f) NPTSR-B. (g) CNNFUS-B. (h) PNN. (i) MHFnet. (j) DBIN. (k) SpfNet.

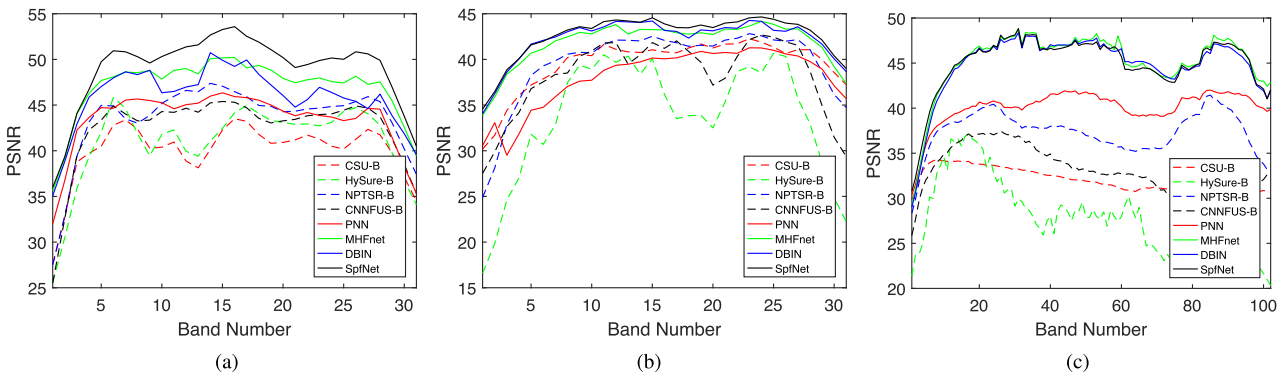


Fig. 9. PSNR as a function of spectral band. (a) CAVE dataset. (b) Harvard dataset. (c) PaviaC dataset.

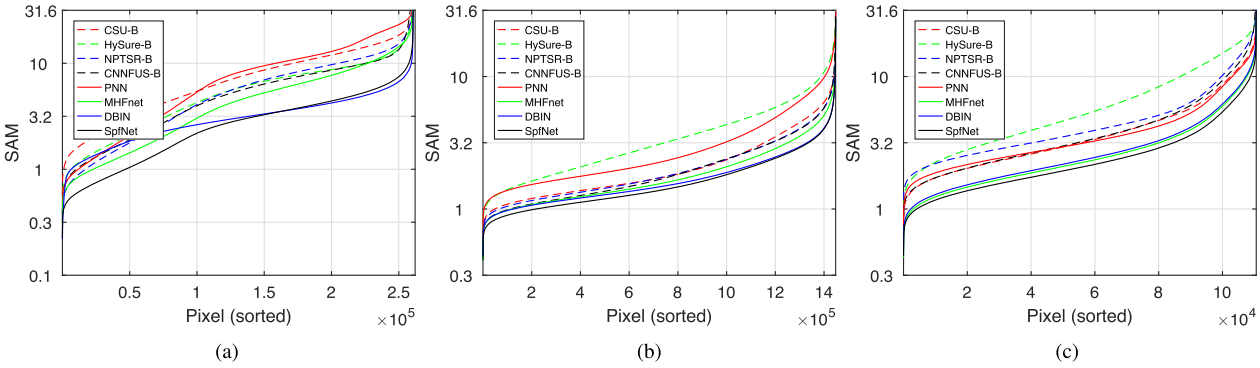


Fig. 10. SAM (plotted in a $\log_{10}(\cdot)$ scale) as a function of sorted pixel. (a) CAVE dataset. (b) Harvard dataset. (c) PaviaC dataset.

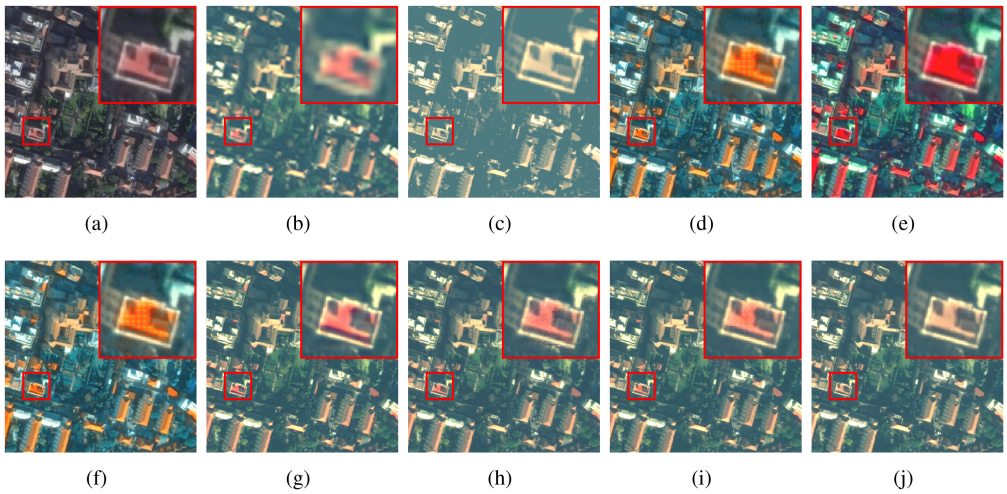


Fig. 11. RGB images (with a meaningful region marked and zoomed in 4 times for easy observation) of HS/MS image fusion results when applied to the WV2 dataset. (a) HR-RGB image. (b) LR-HSI. (c) CSU. (d) HySure. (e) NPTSR. (f) CNNFUS. (g) PNN. (h) MHFnet. (i) DBIN. (j) SpfNet.

part was used for training. Training samples were generated by Wald's protocol, and the input HR-HSI, LR-HSI, and HR-MSI samples are of sizes $32 \times 32 \times 8$, $8 \times 8 \times 8$, and $32 \times 32 \times 3$, respectively. The test stride L was fixed to 4. The spatial blur $\mathbf{B}$ and the spectral response $\mathbf{R}$ of the sensor used in the methods were estimated as in [25]. RGB images of the real dataset and the fusion results of the methods are given in Fig. 11. Visually, it can be see that PNN, MHFnet, DBIN and SpfNet give the good color and brightness results, and the result of the proposed SpfNet is much closer to the HR-RGB image.

V. Conclusion

This article proposed an interpretable patch-aware deep network for HS/MS image fusion by unfolding the subspace-based optimization model. A subspace-based model was built for HS/MS image fusion by performing SVD on each patch of LR-HSI, and two regularization terms were imposed on the model to enforce the target HR-HSI. One regularization term was proposed for pixel localization, and one to extract texture. The complex fusion model was solved by the ADMM algorithm and decoupled into three suboptimization problems. One is associated with the two fidelity terms, which consider spatial-level data fusion and were solved by a gradient descent algorithm. The other two are associated with regularization terms that deal with the injection of detail and were described by proximal operators. Through a deep unfolding technique, the suboptimization problems were represented as three modules, $\mathbf{S}$, $\mathbf{U}$ and $\mathbf{\Pi}$. The $\mathbf{S}$-module implements the basic calculations of the iterative algorithm of the fidelity-based problem, where a linear combination of the gradient functions are replaced by a concatenation operator to provide more flexibility. A u-shaped architecture is used by the $\mathbf{U}$-module to learn the related proximal operator and the $\mathbf{\Pi}$-module uses a reversed u-shaped architecture. A structured deep fusion network was obtained by repeating all steps of the algorithm. To improve the fusion performance, a technique called *aggregation fusion* was proposed. Specifically, to achieve the spectral-spatial fusion, the strategy of image averaging was adopted by convolving and averaging the fused images in all stages of the network. To make full use of redundant information, test images were divided into overlapping patches as input to the network and then aggregated into images. SpfNet has been experimentally tested using three synthetic datasets

and one real dataset. The experimental results demonstrated its effectiveness.

Although the results obtained by the proposed approach are encouraging, there is still large room for further improvements. First, SVD is performed on each patch, and it is better to perform tensor decomposition on patches so that the higher order information of patches can be explored. Second, $\mathbf{S}$-module uses concatenation to relax the fixed format of (3), and a well-defined observation model is more preferable. Third, the structure of the $\mathbf{\Pi}$-module needs to be elaborated and its effectiveness in other scenarios needs to be verified. Fourth, real data have no label information, and it is necessary to extend the network to deal with such a situation that only noisy training samples are available. We will pursue these enhancements in our future research.

ACKNOWLEDGMENT

The authors would like to thank the authors of [19], [23], [25], [30], [39], [61], [69] for providing their code. They would like to thank the anonymous reviewers for their constructive comments on this article.

REFERENCES

[1] D. W. Stein, S. G. Beaven, L. E. Hoff, E. M. Winter, A. P. Schaum, and A. D. Stocker, "Anomaly detection from hyperspectral imagery," *IEEE Signal Process. Mag.*, vol. 19, no. 1, pp. 58–69, Jan. 2002.

[2] H. Van Nguyen, A. Banerjee, and R. Chellappa, "Tracking via object reflectance using a hyperspectral video camera," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, 2010, pp. 44–51.

[3] Z. Pan, G. Healey, M. Prasad, and B. Tromberg, "Face recognition in hyperspectral images," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 25, no. 12, pp. 1552–1560, Dec. 2003.

[4] M. Uzair, A. Mahmood, and A. Mian, "Hyperspectral face recognition with spatio-spectral information fusion and PLS regression," *IEEE Trans. Image Process.*, vol. 24, no. 3, pp. 1127–1137, Mar. 2015.

[5] J. Liu, Z. Wu, J. Li, A. Plaza, and Y. Yuan, "Probabilistic-kernel collaborative representation for spatial-spectral hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 4, pp. 2371–2384, Apr. 2016.

[6] P. Ghamisi *et al.*, "Advances in hyperspectral image and signal processing: A comprehensive overview of the state of the art," *IEEE Geosci. Remote Sens. Mag.*, vol. 5, no. 4, pp. 37–78, Dec. 2017.

[7] S. Li, W. Song, L. Fang, Y. Chen, P. Ghamisi, and J. A. Benediktsson, "Deep learning for hyperspectral image classification: An overview," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 9, pp. 6690–6709, Sep. 2019.

[8] J. Liu, Z. Wu, L. Xiao, and H. Yan, "Learning multiple parameters for kernel collaborative representation classification," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 31, no. 12, pp. 5068–5078, Dec. 2020.

[9] L. Loncan *et al.*, "Hyperspectral pansharpening: A review," *IEEE Geosci. Remote Sens. Mag.*, vol. 3, no. 3, pp. 27–46, Sep. 2015.

[10] N. Yokoya, C. Grohnfeldt, and J. Chanussot, "Hyperspectral and multispectral data fusion: A comparative review of the recent literature," *IEEE Geosci. Remote Sens. Mag.*, vol. 5, no. 2, pp. 29–56, Jun. 2017.

[11] R. Dian, S. Li, B. Sun, and A. Guo, "Recent advances and new guidelines on hyperspectral and multispectral image fusion," *Inf. Fusion*, vol. 69, pp. 40–51, 2021.

[12] L. Alparone, L. Wald, J. Chanussot, C. Thomas, P. Gamba, and L. M. Bruce, "Comparison of pansharpening algorithms: Outcome of the 2006 GRS-S data-fusion contest," *IEEE Trans. Geosci. Remote Sens.*, vol. 45, no. 10, pp. 3012–3021, Oct. 2007.

[13] G. Vivone *et al.*, "A critical comparison among pansharpening algorithms," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 5, pp. 2565–2586, May 2015.

[14] X. Meng, H. Shen, H. Li, L. Zhang, and R. Fu, "Review of the pansharpening methods for remote sensing images based on the idea of meta-analysis: Practical discussion and challenges," *Inf. Fusion*, vol. 46, pp. 102–113, 2019.

[15] G. Vivone *et al.*, "A new benchmark based on recent advances in multispectral pansharpening: Revisiting pansharpening with classical and emerging pansharpening methods," *IEEE Geosci. Remote Sens. Mag.*, vol. 9, no. 1, pp. 53–81, Mar. 2021.

[16] T.-M. Tu, S.-C. Su, H.-C. Shyu, and P. S. Huang, "A new look at IHS-like image fusion methods," *Inf. Fusion*, vol. 2, no. 3, pp. 177–186, 2001.

[17] F. Nencini, A. Garzelli, S. Baronti, and L. Alparone, "Remote sensing image fusion using the curvelet transform," *Inf. Fusion*, vol. 8, no. 2, pp. 143–156, 2007.

[18] C. Ballester, V. Caselles, L. Igual, J. Verdera, and B. Rougé, "A variational model for P XS image fusion," *Int. J. Comput. Vis.*, vol. 69, no. 1, pp. 43–58, 2006.

[19] Y. Xu, Z. Wu, J. Chanussot, and Z. Wei, "Nonlocal patch tensor sparse representation for hyperspectral image super-resolution," *IEEE Trans. Image Process.*, vol. 28, no. 6, pp. 3034–3047, Jun. 2019.

[20] N. Akhtar, F. Shafait, and A. Mian, "Sparse spatio-spectral representation for hyperspectral image super-resolution," in *Proc. Euro. Conf. Comput. Vis.*, Springer, 2014, pp. 63–78.

[21] M. A. Veganzones, M. Simoes, G. Licciardi, N. Yokoya, J. M. Bioucas-Dias, and J. Chanussot, "Hyperspectral super-resolution of locally low rank images from complementary multisource data," *IEEE Trans. Image Process.*, vol. 25, no. 1, pp. 274–288, Jan. 2016.

[22] N. Yokoya, T. Yairi, and A. Iwasaki, "Coupled nonnegative matrix factorization unmixing for hyperspectral and multispectral data fusion," *IEEE Trans. Geosci. Remote Sens.*, vol. 50, no. 2, pp. 528–537, Feb. 2012.

[23] C. Lanaras, E. Baltsavias, and K. Schindler, "Hyperspectral super-resolution by coupled spectral unmixing," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2015, pp. 3586–3594.

[24] R. Dian, S. Li, L. Fang, and Q. Wei, "Multispectral and hyperspectral image fusion with spatial-spectral sparse representation," *Inf. Fusion*, vol. 49, pp. 262–270, 2019.

[25] M. Simões, J. Bioucas-Dias, L. B. Almeida, and J. Chanussot, "A convex formulation for hyperspectral image superresolution via subspace-based regularization," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 6, pp. 3373–3388, Jun. 2015.

[26] Q. Wei, J. Bioucas-Dias, N. Dobigeon, and J.-Y. Tourneret, "Hyperspectral and multispectral image fusion based on a sparse representation," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 7, pp. 3658–3668, Jul. 2015.

[27] W. Dong *et al.*, "Hyperspectral image super-resolution via non-negative structured sparse representation," *IEEE Trans. Image Process.*, vol. 25, no. 5, pp. 2337–2352, May 2016.

[28] R. Dian and S. Li, "Hyperspectral image super-resolution via subspace-based low tensor multi-rank regularization," *IEEE Trans. Image Process.*, vol. 28, no. 10, pp. 5135–5146, Oct. 2019.

[29] J. Liu, Z. Wu, L. Xiao, J. Sun, and H. Yan, "A truncated matrix decomposition for hyperspectral image super-resolution," *IEEE Trans. Image Process.*, vol. 29, pp. 8028–8042, Jul. 2020.

[30] R. Dian, S. Li, and X. Kang, "Regularizing hyperspectral and multispectral image fusion by CNN denoiser," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 32, pp. 1124–1135, Mar. 2021.

[31] S. Li, R. Dian, L. Fang, and J. M. Bioucas-Dias, "Fusing hyperspectral and multispectral images via coupled sparse tensor factorization," *IEEE Trans. Image Process.*, vol. 27, no. 8, pp. 4118–4130, Aug. 2018.

[32] C. I. Kanatsoulis, X. Fu, N. D. Sidiropoulos, and W.-K. Ma, "Hyperspectral super-resolution: A coupled tensor factorization approach," *IEEE Trans. Signal Process.*, vol. 66, no. 24, pp. 6503–6517, Dec. 2018.

[33] R. Dian, S. Li, L. Fang, T. Lu, and J. M. Bioucas-Dias, "Nonlocal sparse tensor factorization for semiblind hyperspectral and multispectral image fusion," *IEEE Trans. Cybern.*, vol. 50, no. 10, pp. 4469–4480, Oct. 2020.

[34] J. Xue, Y. Zhao, Y. Bu, W. Liao, J. Chan, and W. Philips, "Spatial-spectral structured sparse low-rank representation for hyperspectral image super-resolution," *IEEE Trans. Image Process.*, vol. 30, pp. 3084–3097, Feb. 2021.

[35] L. Zhang, L. Song, B. Du, and Y. Zhang, "Nonlocal low-rank tensor completion for visual data," *IEEE Trans. Cybern.*, vol. 51, no. 2, pp. 673–685, Feb. 2021.

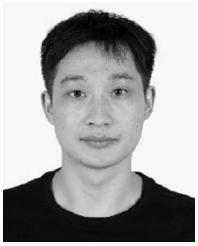
[36] X. X. Zhu *et al.*, "Deep learning in remote sensing: A comprehensive review and list of resources," *IEEE Geosci. Remote Sens. Mag.*, vol. 5, no. 4, pp. 8–36, Dec. 2017.

[37] L. Zhang, L. Zhang, and B. Du, "Deep learning for remote sensing data: A technical tutorial on the state of the art," *IEEE Geosci. Remote Sens. Mag.*, vol. 4, no. 2, pp. 22–40, Jun. 2016.

[38] F. Palsson, J. R. Sveinsson, and M. O. Ulfarsson, "Multispectral and hyperspectral image fusion using a 3-D-convolutional neural network," *IEEE Geosci. Remote Sens. Lett.*, vol. 14, no. 5, pp. 639–643, May 2017.
[39] G. Scarpa, S. Vitale, and D. Cozzolino, "Target-adaptive CNN-based pansharpening," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 9, pp. 5443–5457, Sep. 2018.
[40] Q. Yuan, Y. Wei, X. Meng, H. Shen, and L. Zhang, "A multiscale and multidepth convolutional neural network for remote sensing imagery pan-sharpening," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 11, no. 3, pp. 978–989, Mar. 2018.
[41] Y. Zhang, C. Liu, M. Sun, and Y. Ou, "Pan-sharpening using an efficient bidirectional pyramid network," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 8, pp. 5549–5563, Aug. 2019.
[42] Y. Zheng, J. Li, Y. Li, J. Guo, X. Wu, and J. Chanussot, "Hyperspectral pansharpening using deep prior and dual attention residual network," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 11, pp. 8059–8076, Nov. 2020.
[43] J. Jiang, H. Sun, X. Liu, and J. Ma, "Learning spatial-spectral prior for super-resolution of hyperspectral imagery," *IEEE Trans. Comput. Imag.*, vol. 6, pp. 1082–1096, May 2020.
[44] X. Zhang, W. Huang, Q. Wang, and X. Li, "SSR-NET: Spatial-spectral reconstruction network for hyperspectral and multispectral image fusion," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 7, pp. 5953–5965, Jul. 2021.
[45] X. Dong, X. Sun, X. Jia, Z. Xi, L. Gao, and B. Zhang, "Remote sensing image super-resolution using novel dense-sampling networks," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 2, pp. 1618–1633, Feb. 2021.
[46] D. Liu, J. Li, and Q. Yuan, "A spectral grouping and attention-driven residual dense network for hyperspectral image super-resolution," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 9, pp. 7711–7725, Sep. 2021.
[47] J. Li *et al.*, "Hyperspectral image super-resolution by band attention through adversarial learning," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 6, pp. 4304–4318, Jun. 2020.
[48] W. Dong, S. Hou, S. Xiao, J. Qu, Q. Du, and Y. Li, "Generative dual-adversarial network with spectral fidelity and spatial enhancement for hyperspectral pansharpening," *IEEE Trans. Neural Netw. Learn. Syst.*, to be published, doi: 10.1109/TNNLS.2021.3084745.
[49] J. Yang, X. Fu, Y. Hu, Y. Huang, X. Ding, and J. Paisley, "Pannet: A deep network architecture for pan-sharpening," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 5449–5457.
[50] W. Xie, J. Lei, Y. Cui, Y. Li, and Q. Du, "Hyperspectral pansharpening with deep priors," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 31, no. 5, pp. 1529–1543, May 2019.
[51] X. Fu, W. Wang, Y. Huang, X. Ding, and J. Paisley, "Deep multiscale detail networks for multiband spectral image sharpening," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 32, no. 5, pp. 2090–2104, May 2021.
[52] L.-J. Deng, G. Vivone, C. Jin, and J. Chanussot, "Detail injection-based deep convolutional neural networks for pansharpening," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 8, pp. 6995–7010, Aug. 2021.
[53] R. Dian, S. Li, A. Guo, and L. Fang, "Deep hyperspectral image sharpening," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 11, pp. 5345–5355, Nov. 2018.
[54] H. Shen, M. Jiang, J. Li, Q. Yuan, Y. Wei, and L. Zhang, "Spatial-spectral fusion by combining deep learning and variational model," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 8, pp. 6169–6181, Aug. 2019.
[55] L. Zhang, J. Nie, W. Wei, Y. Li, and Y. Zhang, "Deep blind hyperspectral image super-resolution," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 32, pp. 2388–2400, Jun. 2021.
[56] J. R. Hershey, J. L. Roux, and F. Weninger, "Deep unfolding: Model-based inspiration of novel deep architectures," 2014, *arXiv:1409.2574*.
[57] K. H. Jin, M. T. McCann, E. Froustey, and M. Unser, "Deep convolutional neural network for inverse problems in imaging," *IEEE Trans. Image Process.*, vol. 26, no. 9, pp. 4509–4522, Sep. 2017.
[58] J. Ma, X.-Y. Liu, Z. Shou, and X. Yuan, "Deep tensor ADMM-net for snapshot compressive imaging," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2019, pp. 10223–10232.
[59] Y. Yang, J. Sun, H. Li, and Z. Xu, "ADMM-CSNet: A deep learning approach for image compressive sensing," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 3, pp. 521–538, Mar. 2020.
[60] H. Gupta, K. H. Jin, H. Q. Nguyen, M. T. McCann, and M. Unser, "CNN-based projected gradient descent for consistent CT image reconstruction," *IEEE Trans. Med. Imag.*, vol. 37, no. 6, pp. 1440–1453, Jun. 2018.
[61] Q. Xie, M. Zhou, Q. Zhao, D. Meng, W. Zuo, and Z. Xu, "Multispectral and hyperspectral image fusion by MS/HS fusion net," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 1585–1594.
[62] K. Zhang, L. V. Gool, and R. Timofte, "Deep unfolding network for image super-resolution," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 3217–3226.
[63] W. Wei, J. Nie, Y. Li, L. Zhang, and Y. Zhang, "Deep recursive network for hyperspectral image super-resolution," *IEEE Trans. Comput. Imag.*, vol. 6, pp. 1233–1244, Aug. 2020.
[64] J. Zhang and B. Ghanem, "ISTA-Net: Interpretable optimization-inspired deep network for image compressive sensing," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 1828–1837.
[65] D. Shen, J. Liu, Z. Xiao, J. Yang, and L. Xiao, "A twice optimizing net with matrix decomposition for hyperspectral and multispectral image fusion," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 13, pp. 4095–4110, Jul. 2020.
[66] J. Liu, Z. Wu, L. Xiao, and X.-J. Wu, "Model inspired autoencoder for unsupervised hyperspectral image super-resolution," 2021, *arXiv:abs/2110.11591*.
[67] W. Dong, C. Zhou, F. Wu, J. Wu, G. Shi, and X. Li, "Model-guided deep hyperspectral image super-resolution," *IEEE Trans. Image Process.*, vol. 30, pp. 5754–5768, May 2021.
[68] D. Shen, J. Liu, Z. Wu, J. Yang, and L. Xiao, "ADMM-HFNet: A matrix decomposition-based deep approach for hyperspectral image fusion," *IEEE Trans. Geosci. Remote Sens.*, to be published, doi: 10.1109/TGRS.2021.3112181.
[69] W. Wang, W. Zeng, Y. Huang, X. Ding, and J. Paisley, "Deep blind hyperspectral image fusion," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2019, pp. 4150–4159.
[70] R. C. Hardie, M. T. Eismann, and G. L. Wilson, "MAP estimation for hyperspectral image resolution enhancement using an auxiliary sensor," *IEEE Trans. Image Process.*, vol. 13, no. 9, pp. 1174–1184, Sep. 2004.
[71] J. M. Nascimento and J. M. Dias, "Vertex component analysis: A fast algorithm to unmix hyperspectral data," *IEEE Trans. Geosci. Remote Sens.*, vol. 43, no. 4, pp. 898–910, Apr. 2005.
[72] X. Wang, R. Girshick, A. Gupta, and K. He, "Non-local neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 7794–7803.
[73] A. Vaswani *et al.*, "Attention is all you need," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 5998–6008.
[74] Q. Wei, N. Dobigeon, and J.-Y. Tourneret, "Fast fusion of multi-band images based on solving a Sylvester equation," *IEEE Trans. Image Process.*, vol. 24, no. 11, pp. 4109–4121, Nov. 2015.
[75] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional networks for biomedical image segmentation," in *Proc. Int. Conf. Med. Image Comput. Comput.-Assist. Interv.*, 2015, pp. 234–241.
[76] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 770–778.
[77] D. Ulyanov, A. Vedaldi, and V. Lempitsky, "Deep image prior," *Int. J. Comput. Vis.*, vol. 128, no. 7, pp. 1867–1888, 2020.
[78] K. C. Mertens, B. De Baets, L. P. Verbeke, and R. R. De Wulf, "A sub-pixel mapping algorithm based on sub-pixel/pixel spatial attraction models," *Int. J. Remote Sens.*, vol. 27, no. 15, pp. 3293–3310, 2006.
[79] F. Yasuma, T. Mitsunaga, D. Iso, and S. Nayar, "Generalized assorted pixel camera: Post-capture control of resolution, dynamic range and spectrum," *IEEE Tran. Image Process.*, vol. 19, no. 9, Mar. 2010.
[80] A. Chakrabarti and T. Zickler, "Statistics of real-world hyperspectral images," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2011, pp. 193–200.
[81] T. Ranchin and L. Wald, "Fusion of high spatial and spectral resolution images: The Arsis concept and its implementation," *Photogrammetric Eng. Remote Sens.*, vol. 66, no. 1, pp. 49–61, 2000.

Jianjun Liu (Member, IEEE) received the B.S. degree in applied mathematics and the Ph.D. degree in pattern recognition and intelligence system from the Nanjing University of Science and Technology, Nanjing, China, in 2009 and 2014, respectively.

He is currently an Associate Professor with the School of Artificial Intelligence and Computer Science, Jiangnan University, Wuxi, China. From 2018 to 2020, he was a Postdoctoral Researcher with the Department of Electrical Engineering, City University of Hong Kong, Hong Kong. His research interests include hyperspectral image classification, super-resolution, spectral unmixing, sparse representation, computer vision, and pattern recognition.



Dunbin Shen received the B.S. degree in digital media technology from Hubei Minzu University, Enshi City, China, in 2016, and the M.S. degree in computer science and technology from Jiangnan University, Wuxi, China, in 2021. He is currently working toward the Ph.D. degree in signal and information processing with the School of Information and Communication Engineering, Dalian University of Technology, Dalian, China.

His research interests include hyperspectral image fusion, deep learning, and hyperspectral target detection.

Zebin Wu (Senior Member, IEEE) received the B.S. and Ph.D. degrees in computer science from the Nanjing University of Science and Technology, Nanjing, China, in 2003 and 2008, respectively.

He is currently a Professor with the School of Computer Science, Nanjing University of Science and Technology. His research interests include hyperspectral image processing, high-performance computing, and computer simulation.

Liang Xiao (Member, IEEE) received the B.S. degree in applied mathematics and the Ph.D. degree in computer science from the Nanjing University of Science and Technology, Nanjing, China, in 1999 and 2004, respectively.

From 2009 to 2010, he was a Postdoctoral Fellow with Rensselaer Polytechnic Institute, Troy, NY, USA. He is currently a Professor with the School of Computer Science, Nanjing University of Science and Technology. His main research interests include inverse problems in image processing, scientific computing, data mining, and pattern recognition.

Jun Sun (Member, IEEE) received the Ph.D. degree in control theory and engineering, and the M.Sc. degree in computer science and technology from Jiangnan University, Wuxi, China, in 2009 and 2003, respectively.

He is currently a Full Professor with the School of Artificial Intelligence and Computer Science, Jiangnan University. He is also the Vice Director of Jiangsu Provincial Engineering Laboratory of Pattern Recognition and Computational Intelligence, Jiangsu Province. He has authored or coauthored more than 150 papers in journals, conference proceedings, and several books in the area of his research interests. His major research interests include computational intelligence, machine learning, bioinformatics, among others.

Hong Yan (Fellow, IEEE) received the B.S. degree from the Nanjing University of Posts and Telecommunications, Nanjing, China, in 1982, the M.S. degree from the University of Michigan, Ann Arbor, MI, USA, in 1984, and the Ph.D. degree from Yale University, New Haven, CT, USA, in 1989, all in electrical engineering.

From 1986 to 1989, he was a Research Scientist with General Network Corporation, New Haven, CT, USA, where he researched on design and optimization of computer and telecommunications networks. He joined the University of Sydney, Sydney, NSW, Australia, in 1989, and became a Professor of Imaging Science in 1997. He is currently a Professor of Electrical Engineering with the City University of Hong Kong, Hong Kong. He has over 600 publications in these areas of his research interests. His research interests include image processing, pattern recognition, and bioinformatics.

Prof. Yan was elected as an IAPR Fellow for contributions to document image analysis and an IEEE Fellow for contributions to image recognition techniques and applications. He was the recipient of the 2016 Norbert Wiener Award from the IEEE SMC Society for contributions to image and biomolecular pattern recognition techniques.