Joint Sparse Representation-based Single Image Super-Resolution for Remote Sensing Applications

Bhabesh Deka, *Senior Member, IEEE*, Helal Uddin Mullah, Trishna Barman, and Sumit Datta, *Member, IEEE*

***Abstract*—Sparse representation-based single image super-resolution (SISR) methods use a coupled overcomplete dictionary trained from high-resolution images/image patches. Since remote sensing (RS) satellites capture images of large areas, these images usually have poor spatial resolution and obtaining an effective dictionary as such would be very challenging. Moreover, traditional patch-based sparse representation models for reconstruction tend to give unstable sparse solution and produce visual artefact in the recovered images. To mitigate these problems, in this article, we have proposed an adaptive joint sparse representation-based SISR method that is dependent only on the input low-resolution image for dictionary training and sparse reconstruction. The new model combines patch-based local sparsity and group sparse representation-based nonlocal sparsity in a single framework, which helps in stabilizing the sparse solution and improve the SISR results. The experimental results are evaluated both visually and quantitatively for several RGB and multispectral RS datasets, where the proposed method shows improvements in peak signal-to-noise ratio by 1–4 dB and 2–3 dB over the state-of-the-art sparse representation- and deep learning-based SR methods, respectively. Land cover classification applied on the super-resolved images further validate the advantages of the proposed method. Finally, for practical RS applications, we have performed parallel implementation in general purpose graphics processing units and achieved significant speed ups (30–40×) in the execution time.**

***Index Terms*—Dictionary training, joint sparse representation (JSR), parallel processing, remote sensing (RS), super-resolution.**

I. Introduction

REMOTE sensing (RS) applications, such as surveillance, change detection, disaster management, and agriculture monitoring, require images with high spatial and spectral resolutions. High spatial information allows the accurate geometric analysis, while high spectral resolution allows better thematic interpretation. With recent developments in high-definition displays, there are numerous applications, where the existing low-resolution (LR) images are not preferred. Although high-resolution (HR) images can be acquired by using state-of-the-art HR imaging sensors directly, they are expensive in implementation. Moreover, an LR satellite sensor that is already in the orbit cannot be replaced with a new one. So, reconstruction of HR images from the available LR images by signal processing becomes an important and highly promising topic of research [1], [2], [3], [4], [5].

Single image super-resolution (SISR) is considered as an ill-posed inverse problem in image processing, where an HR image is to be restored from one of its available LR versions. The traditional SISR techniques apply standard interpolation techniques to estimate the missing pixels in the target HR image. They usually produce low-quality images with artifacts at higher zooming factors as they do not consider *a priori* information of the target HR image [6]. Regularization-based methods exploit specific prior information about the target HR image, e.g., the sparsity of the image gradient magnitudes (total variation-norm) [7] or gradient profile information [8] as constraints along with the least-square-based data fidelity term. Recently, dictionary learning and sparse representation have been successfully applied to natural image SR [9]. Their success owes to the fact that natural images are intrinsically sparse in one or more domains, and to solve the underlying sparse representation problem, there are highly efficient ℓ_1-solvers [10], [11], [12]. Sparse representation can be unveiled by efficient dictionaries, which are constructed either from a fixed basis set (e.g., overcomplete DCT, wavelets, etc.) or learned iteratively from an external dataset/the test image itself [13], [14].

In this article, we focus on exploiting nonlocal similarity among patches to form patch groups and then formulate a JSR problem for SISR of RS images. It is verified that similar patches do occur multiple times within a single image irrespective of image scaling [15]. First, we train an adaptive dictionary (patch-based) from the single LR image to overcome the traditional dictionary training problem from an external database. This is very significant for RS applications as relevant satellites provide images only at low spatial resolution. Moreover, as the dictionary training is usually carried out offline and before sparse reconstruction, so the same would not fit well for different test images. Hence, in the proposed SISR, we learn an adaptive dictionary exclusively from patches taken from the given LR RS image, and then the corresponding HR image is reconstructed. The adaptive learning has obvious advantageous as the dictionary atoms are more correlated to the input image patches [16]. It can be deployed for real RS applications, as it is independent

Manuscript received 25 September 2022; revised 4 December 2022 and 25 December 2022; accepted 27 January 2023. Date of publication 16 February 2023; date of current version 2 March 2023. This work was supported in part by ISRO under the RESPOND Project ISRO/RES/4/642/17-18 and in part by Digital India Corporation, Ministry of Electronics and Information Technology (MeiTY), GoI, under the Visvesvaraya Ph.D. Scheme (Ph.D./MLA/04(41)/2015-16/01) for providing financial supports, which helped in smooth conduction of the above research work. *(Corresponding author: Bhabesh Deka.)*

Bhabesh Deka, Helal Uddin Mullah, and Trishna Barman are with the Department of Electronics and Communication Engineering, Tezpur University, Tezpur 784028, India (e-mail: bdeka@tezu.ernet.in; hmullah@tezu.ernet.in; trish90@tezu.ernet.in).

Sumit Datta is with the School of Electronic Systems and Automation, Digital University Kerala, Thiruvananthapuram 695317, India (e-mail: sumit.datta@iiitmk.ac.in).

Digital Object Identifier 10.1109/JSTARS.2023.3244069

of any external dataset. Furthermore, to ensure quality of the trained dictionary, we have selected only key patches from the training patch dataset by applying a variance-based patch pruning technique.

In particular, we propose a novel JSR-based adaptive SISR reconstruction by integrating both patch and group sparsity (PSR and GSR) in a common framework. We solve the above two subtasks using the alternating direction method of multipliers (ADMMs) approach. The role of PSR subproblem is to reduce the oversmoothing effects by inducing precise local information (through the patch sparsity constraint), while that of GSR subproblem is to subdue the visual artifacts of final image by maintaining nonlocal uniformity (through the nonlocal similarity constraint). Next, we consider CUDA-GPU-based parallel processing for accelerating the proposed algorithm intended for practical RS applications. In a nutshell, the major contributions of this work are listed as follows.

1) A new adaptive SISR algorithm is developed using the JSR strategy for SR of LR RS images. The proposed algorithm utilizes a PSR subproblem for restoring local patch-based high-frequency information and a GSR subproblem for inducing the nonlocal information, which further refines the PSR reconstructed images. Then, the two subproblems are fed into an ADMM framework to obtain the final HR reconstructed image.
2) Adaptive patch-based coupled overcomplete dictionary is learnt from the single test image itself using the K-SVD algorithm, thereby making it independent of any external training dataset. Furthermore, nonlocal similarity-based multiple adaptive group dictionaries are learnt for each patch group using the rank minimization technique.

The rest of this article is organized as follows. In Section II, the related SR works are briefly discussed. A brief background on the sparse and group sparse representation (GSR) techniques are discussed in Section III. In Section IV, the proposed method is explained across two sections, namely, adaptive patch- and group-dictionary training, and joint sparse reconstruction-based SR. Experimental setup and detailed simulation results are presented in Section V. Section VI provides a brief discussion on the simulation results. Finally, Section VII concludes this article.

II. Related Works

Yang et al. [14] proposed a sparse coding-based SR (ScSR) technique for natural images with a coupled dictionary learned from an external dataset [17]. Apart from computational cost, it is highly dependent on the quality of the trained dictionary and SR results are not up to the mark for highly textured RS images. Moustafa et al. [17] proposed self-example learning and sparse representation-based multispectral (MS) SR using morphological features. This method not only provides a fast implementation using parallel processing but also avoids external data dependency for dictionary training. However, it ignores nonlocal similarity of image patches within or across different scales of an image, which is widely considered to be a strong reconstruction constraint. Convolutional sparse coding (CSC) was used to perform image SR, where sparse coding via linear combinations of a set of dictionary vectors (as adopted in ScSR) is replaced by the sum of a set of convolutions with dictionary filters [18]. Furthermore, Song et al. [19] proposed the multimodal image SR algorithm known as "CDLSR," where images of different modalities in the sparse feature domain are coupled to learn a set of dictionaries followed by a two-stage reconstruction of the target HR image via joint sparse representations (JSRs), given another related image modality. Romano et al. [4] proposed the accurate image SR technique utilizing a set of prelearned filters from image patches, which is able to restore a sharper and contrast enhanced image.

Recently, a collaborative sparse representation-based regularization along with nonlocal self-similarity (CRNS) is proposed by Chang et al. [20], which is able to provide effective recovery of high-frequency information and minimum error for the SR of natural images. Hou et al. [21] proposed a global joint dictionary learning model-based RS SR exploiting nonlocal self-similarity (NLSS) and sparsity constraints. However, they have shortcomings, such as point-like targets, and complex textures not being fairly discernible in the output.

In patch-based sparse coding, individual image patches are reconstructed separately assuming that they are independent and uncorrelated due to which instabilities among sparse solutions arise, and hence, artifacts are visible in the reconstructed image. However, the recent SR methods focus on the group sparse representation of similar patches to maintain consistency among the recovered patches [22], [23], [24], [25], [26], [27]. Xu and Gao [25] applied the group-sparsity concept of Zhang et al. [23] to perform natural image SR by utilizing the Gaussian distance to measure similarity and named it as "GSRGSiSR." In another work, Yang et al. [24] proposed group-structured sparse representation-based SR method by forming clusters of similar nonlocal patches and then sparse coding with principal component analysis-based online dictionary learning. Lu et al. [28] proposed a group-based SISR with online dictionary learning, where they basically consider a patch group instead of a single patch. Although the above group-based methods are able to slightly reduce the reconstruction error, yet there is ample scope for further improvements.

Researchers have also shown interests to combine the sparsity model with other image refinement constraints, such as nonlocal self similarity (NLSS) based group sparsity to develop a JSR model. A joint patch-group-based sparse representation model was proposed by Zha et al. [29] for the restoration (e.g., inpainting or deblocking) of natural (photographic) images. Mikaeli et al. [30] proposed a patch- and group-based local smoothness modeling prior for SISR. Gao et al. [31] demonstrated an SR method using JSR and low-rank approximation to enhance the spectral information of MS images. This method indirectly performs spatial SR for the hyperspectral images up to a level of MS image's resolution. In another work [32], Zha et al. proposed a low-rank regularized group sparse coding (LR-GSC) based image restoration method. They have demonstrated better results than a few existing methods for different tasks, such as image inpainting and compressive sensing. Shi et al. [33] demonstrated a joint regularization problem involving patch-based and pixel-based models for face image hallucination preserving local

details and structures present in the images. Liu et al. [34] proposed an adaptive joint distribution modeling-based SR method exploiting the gradient sparsity and nonlocal patch correlation. However, their results are still lacking in peak signal-to-noise ratio (PSNR) even for high-quality datasets (e.g., Set5 and Set14).

Recently, deep learning (DL) based methods have become very popular in the field of image SR due to their ability to synthesize the missing data by learning a mapping function from LR to HR images. Wang et al. [35] developed a lightweight SR network that incorporates context enhancement and contextual feature aggregation modules to boost the representational power of the extracted feature. Dong et al. [36] proposed a practical degradation model and a kernel-aware SR network for real-world RS images in order to address the degradation and the high-frequency recovery issues. Lei and Shi [37] presented a hybrid-scale self-similarity exploitation network that enhances feature representations by exploiting single- and cross-scale similarity information in RS images. SR methods using convolutional neural network (CNN) and generative adversarial networks (GANs) are mostly explored for the generation of HR images [38], [39], [40], [41], [42], [43]. These methods are generally data intensive and often require retraining (offline) of the existing models whenever data from a new sensor with different spatial resolutions are encountered. Since RS satellites acquire MS images at low spatial resolution, collection of a large dataset becomes practically difficult, and hence, the model performance is severely affected. It is also observed that the sparse representation-based methods are computationally expensive because of iterative minimization steps as well as several high-dimensional matrix multiplications involving the dictionaries and image data. Tan et al. [44] applied LASSO for sparse approximation and CUDA-based parallel implementation to achieve 30–35× speed up. A similar work is presented by Attarde and Khaparde [45] for natural image SR.

III. Prior Art

A. Formulation of PSR and GSR

An image patch $\mathbf{x}_i \in \mathbb{R}^{m\times 1} \forall i = 1, 2, \ldots, N$ can be approximated as a linear combination of a few selected basis functions or atoms/columns of an overcomplete dictionary $\mathbf{D} \in \mathbb{R}^{m\times n}$, where $m \ll n$ and the sparse coefficient vector $\boldsymbol{\alpha}_i \in \mathbb{R}^{n\times 1}$, i.e.,

$$\mathbf{x}_i = \mathbf{D}\boldsymbol{\alpha}_i \tag{1}$$

where the symbols *m*, *n* and *N* are used to represent sizes of image patch (i.e.,$\sqrt{m} \times \sqrt{m}$), dictionary atoms, and the total number of patches in the image, respectively.

A schematic representation of the PSR formulation is depicted in Fig. 1(a). The solution of (1), an underdetermined system of equations, can be approximated as follows:

$$\hat{\boldsymbol{\alpha}}_i = \arg\min_{\boldsymbol{\alpha}_i} \left(\tfrac{1}{2}\left\|\mathbf{x}_i - \mathbf{D}\boldsymbol{\alpha}_i\right\|_2^2 + \lambda \left\|\boldsymbol{\alpha}_i\right\|_1\right) \tag{2}$$

where λ is a constant, known as the regularization parameter. Here, the dictionary $\mathbf{D}$ is shared by all the patches extracted from the given image. So, if we concatenate N such patches extracted from the image as $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_N] \in \mathbb{R}^{m\times N}$, then we can rewrite (2) as follows:

$$\hat{\boldsymbol{\alpha}} = \arg\min_{\boldsymbol{\alpha}} \left(\frac{1}{2}\left\|\mathbf{X} - \mathbf{D}\boldsymbol{\alpha}\right\|_2^2 + \lambda \left\|\boldsymbol{\alpha}\right\|_1\right) \tag{3}$$

where $\boldsymbol{\alpha} = [\boldsymbol{\alpha}_1, \ldots, \boldsymbol{\alpha}_N] \in \mathbb{R}^{n\times N}$ is the sparse coefficient matrix and ℓ_1-norm is applied to each column of $\boldsymbol{\alpha}$ corresponding to each patch in $\mathbf{X}$.

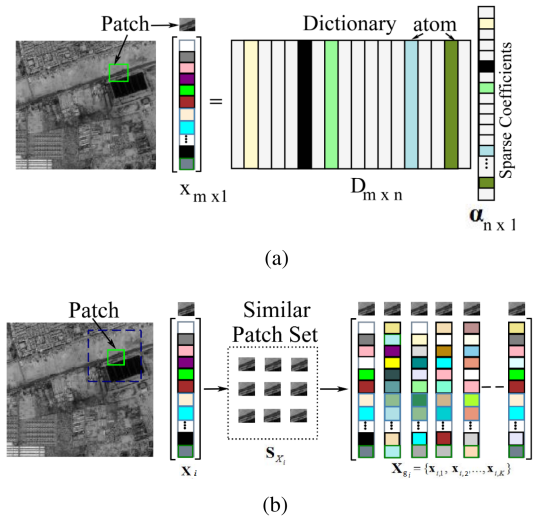


Fig. 1. Schematic representation of patchwise sparse representation (top) and formation of patch-group matrix (bottom) (source: [23], Fig. 1). (a) Patch-based sparse representation. (b) Patch-group formation.

As shown in Fig. 1(b), a patch group may be represented by a matrix with columns formed by nonlocal image patch vectors having the most identical features in them. For each example patch $\mathbf{x}_i$, the corresponding most similar k patches are selected from a search window of size $W \times W$. Here, the similarity is measured in terms of the Euclidean distance between the target patch $\mathbf{x}'$ and the candidate patch $\mathbf{x}$ as follows:

$$d_i = \sqrt{\sum_{i=1}^{\sqrt{m}} \sum_{j=1}^{\sqrt{m}} \left(x_{i,j} - x'_{i,j}\right)^2} \tag{4}$$

where $x_{i,j}$ and $\acute{x}_{i,j}$ represent the pixel values of the target and candidate patches, respectively, while d_i is the distance between them. Now, similar patches $S_{\mathbf{x}_i}$ within the search window $W \times W$ are vectorized and stacked as columns of the group matrix $\mathbf{X}_{g_i} = (\mathbf{x}_{(i,1)}, \ldots, \mathbf{x}_{(i,k)}) \in \mathbb{R}^{m\times k}$, where $\mathbf{x}_{(i,k)}$ represents the kth patch in the ith group. Similar to (2), assuming a group dictionary $\mathbf{D}_{g_i} \in \mathbb{R}^{m\times M}$ having M atoms, we can carry out sparse coding for each patch group $\mathbf{X}_{g_i}$ corresponding to the ith patch over $\mathbf{D}_{g_i}$ as follows:

$$\hat{\boldsymbol{\beta}}_{g_i} = \arg\min_{\boldsymbol{\beta}_{g_i}} \left(\frac{1}{2}\left\|\mathbf{X}_{g_i} - \mathbf{D}_{g_i}\boldsymbol{\beta}_{g_i}\right\|_2^2 + \lambda \left\|\boldsymbol{\beta}_{g_i}\right\|_1\right) \tag{5}$$

where $\boldsymbol{\beta}_{g_i} \in \mathbb{R}^{M\times k}$ are the corresponding group sparse coefficients' matrices. If we put all the patch groups extracted from N patch locations of the given image $\mathbf{X}$ together as $\mathbf{X}_G = (\mathbf{X}_{g_1}, \mathbf{X}_{g_2}, \ldots, \mathbf{X}_{g_N}) \in \mathbb{R}^{m\times kN}$, we can use a combined group dictionary $\mathbf{D}_G = (\mathbf{D}_{g_1}, \mathbf{D}_{g_2}, \ldots, \mathbf{D}_{g_N}) \in \mathbb{R}^{m\times MN}$, and then

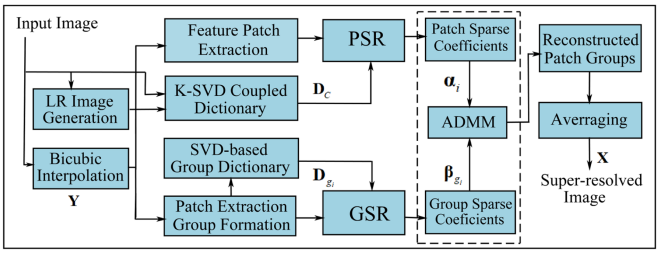


Fig. 2. Schematic of the proposed JSR-based SISR method.

rewrite (5) as follows:

$$\hat{\boldsymbol{\beta}}_G = \arg\min_{\boldsymbol{\beta}_G} \left(\frac{1}{2} \left\| \mathbf{X}_G - \mathbf{D}_G \boldsymbol{\beta}_G \right\|_2^2 + \lambda \left\| \boldsymbol{\beta}_G \right\|_1 \right) \tag{6}$$

where $\boldsymbol{\beta}_G \in \mathbb{R}^{MN \times kN}$ is an expanded version of $\boldsymbol{\beta}_{g_i}$.

IV. Proposed Method

The proposed algorithm restores the desired HR image by solving two sparse coding subproblems—the PSR and the GSR for individual image patches and patch groups, respectively. These subproblems can be combined for the composite JSR problem, which can be solved using the ADMM framework. We named it as the joint sparse representation-based adaptive single image SR (JASISR). A graphical representation of the proposed SR scheme is shown in Fig. 2. Detailed adaptive dictionary learning and joint sparse reconstruction techniques are explained in the following sections.

A. Adaptive Patch- and Group-Dictionary Learning

The SISR problem, i.e., recovering the HR version $\mathbf{X}$ of an image from the observed LR image $\mathbf{Y}$, is defined as follows:

$$\mathbf{Y} = SH\mathbf{X} + \mathbf{n} \tag{7}$$

where S is the downsampling factor, H is the blurring operator, and $\mathbf{n}$ is the additive white Gaussian noise or representation error, which is taken care of by the fidelity term while solving a sparse approximation problem. Two adaptive dictionaries are learned from $\mathbf{Y}$, one for the PSR and the other for the GSR. First, we learn a coupled overcomplete dictionary $\mathbf{D}_C$ from the HR–LR combined patch dataset $\mathbf{Y}_C = (\mathbf{Y}_h; \mathbf{Y}_\ell)$. Here, HR patches with the most relevant information are directly extracted from the input image itself and stored as columns in the matrix $\mathbf{Y}_h$. On the other hand, for obtaining the corresponding LR feature patch, the given image is first blurred, downsampled, and then resized to the original image size by using bicubic interpolation. Then, the training set is prepared from the key patches having variance values above a certain threshold, while the nonsignificant patches are not included. In order to extract high-frequency features and their orientations from the LR patch, we apply four one-dimensional Sobel filters in the x-, y-, and two diagonal directions ($+45^\circ$ and -45°) on each patch. We prefer the Sobel filters here since they are simple to use and capable of generating effective gradient maps, which essentially import details, such as edges and contours, for accurate sparse representation. Finally, an LR feature vector is formed corresponding to each patch location by concatenating four gradient maps, and then the feature matrix $\mathbf{Y}_\ell$ is formed by stacking LR feature vectors, corresponding to all key patches, as its columns. Assuming that LR and HR dictionaries $\mathbf{D}_h$ and $\mathbf{D}_\ell$ share similar sparse coefficients vectors in $\mathbf{A}$, the coupled dictionary $\mathbf{D}_C = (\mathbf{D}_h; \mathbf{D}_\ell)$ can now be jointly learned from $\mathbf{Y}_C$. Mathematically, the relevant optimization problem is written as follows:

$$\min_{\{\mathbf{D}_C, \mathbf{A}\}} \left\| \mathbf{Y}_C - \mathbf{D}_C \mathbf{A} \right\|_2^2 + \lambda \left(\frac{1}{p} + \frac{1}{q} \right) \left\| \mathbf{A} \right\|_1 \tag{8}$$

where p and q are the dimensions of HR and LR patch vectors, respectively, and λ is the regularization parameter. Equation 8 is efficiently solved by the existing coupled K-SVD algorithm [46]. Next, for the GSR, we learn self-adaptive group dictionaries $\mathbf{D}_{g_i}$ for the individual patch groups $\mathbf{Y}_{g_i}$ rather than learning a single overcomplete dictionary $\mathbf{D}_g$. Because, learning a single dictionary by joint sparse coding of all available patch groups at a time is computationally expensive and may no preserve the individual group specific identity very well. Adaptive group dictionaries $\mathbf{D}_{g_i} (i = 1, 2, \ldots, M)$ are directly learned from the given image patch groups $\mathbf{Y}_{g_i} (i = 1, 2, \ldots, M)$ using rank minimization to obtain its low-rank estimate $\mathbf{Y}_{g_r}$, i.e.,

$$\mathbf{Y}_{g_r} = \mathbf{U}_{g_i} \sum\nolimits_{g_i} \mathbf{V}_{g_i}^T = \sum_{j=1}^{r} \sigma_{g_i \otimes j} \left(\mathbf{u}_{g_i \otimes j} \mathbf{v}_{g_i \otimes j}^T \right) \tag{9}$$

where $\mathbf{U}_{g_r}$ and $\mathbf{V}_{g_r}$ are the orthogonal matrices consisting of the left- and right-singular vectors of $\mathbf{Y}_{g_r}$, respectively, and $\sum_{g_r} = \text{diag}(\gamma_{Y_{g_i \otimes j}})$ is a diagonal matrix consisting of r nonzero singular values of $\mathbf{Y}_{g_i}$ represented by $\gamma_{Y_{g_i \otimes j}} = [\sigma_{g_i \otimes 1}; \sigma_{g_i \otimes 2}; \ldots; \sigma_{g_i \otimes r}]$ on its principal diagonal in the descending order, i.e., $\sigma_{g_i \otimes 1} \geq \sigma_{g_i \otimes 2} \geq \cdots \geq \sigma_{g_i \otimes r}$. Now, if columns of $\mathbf{U}_{g_i}$ and $\mathbf{V}_{g_i}$ are denoted by $\mathbf{u}_{g_i}$ and $\mathbf{v}_{g_i}$, respectively, then each submatrix $\mathbf{d}_{g_i} \in \mathbb{R}^{m \times k}$ of the group dictionary $\mathbf{D}_{g_i}$ is obtained as follows:

$$\mathbf{d}_{g_i \otimes j} = \mathbf{u}_{g_i \otimes j} \mathbf{v}_{g_i \otimes j}^T, \quad j = 1, 2, \ldots, r. \tag{10}$$

The final adaptively learned dictionary for the group $\mathbf{Y}_{g_i}$ can be formulated by concatenating all the r submatrices obtained from above as follows:

$$\mathbf{D}_{g_i} = \left(\mathbf{d}_{g_i \otimes 1}, \mathbf{d}_{g_i \otimes 2}, \ldots, \mathbf{d}_{g_i \otimes r} \right). \tag{11}$$

The proposed self-adaptive group dictionary training is more robust and effective as it assures that all the patches in each group $\mathbf{Y}_{g_i}$ are using the same dictionary $\mathbf{D}_{g_i}$ and sharing the same dictionary atoms. The learning is also very efficient as it needs to solve only one SVD problem for each group.

B. Joint Sparse Reconstruction-Based SR

As shown in Fig. 2, the proposed JSR-based reconstruction first upscales the LR image $\mathbf{Y}$ to a size equal to that of the target HR image using bicubic interpolation. Then, a joint regularization problem incorporating the two subproblems: PSR and GSR is solved using the ADMM technique. Mathematically, it

is given as follows:

$$\begin{aligned}\hat{\mathbf{X}} &= \arg\min_{\mathbf{X}} \|\mathbf{Y} - SH\mathbf{X}\|_2^2 + \mu \Re_{\text{JSR}}(\mathbf{X}) \\ &= \arg\min_{\mathbf{X}} \|\mathbf{Y} - SH\mathbf{X}\|_2^2 + \mu_1 \Re_{\text{PSR}}(\mathbf{X}) + \mu_2 \Re_{\text{GSR}}(\mathbf{X})\end{aligned} \tag{12}$$

where $\Re_{\text{JSR}}(\mathbf{X})$ represents the JSR regularization, which is further split into PSR and GSR terms; μ_1 and μ_2 are the regularization parameters. $\Re_{\text{PSR}}(\mathbf{X})$ solves a patchwise optimization problem using the input bicubic upscaled image $\mathbf{Y}$ and pretrained coupled LR and HR dictionaries obtained from (8), as follows:

$$\Re_{\text{PSR}}(\mathbf{X}) = \sum_i \left(\frac{1}{2} \left\| \tilde{\mathbf{y}}_i - \tilde{\mathbf{D}} \alpha_i \right\|_2^2 + \lambda_1 \|\boldsymbol{\alpha}_i\|_1 \right) \tag{13}$$

where $\tilde{\mathbf{D}} = \begin{pmatrix} P\mathbf{D}_l \\ R\mathbf{D}_h \end{pmatrix}$, $\tilde{\mathbf{y}}_i = \begin{pmatrix} \mathbf{y}_i \\ \gamma \mathbf{w} \end{pmatrix}$, and $\mathbf{y}_i = P(SH\mathbf{X})$ give the ith feature patch extracted by the operator P and $\mathbf{w}$ contains the overlapping pixels between the present HR patch and previously generated HR image. The operator R is used to select the region of overlaps. The features sign search algorithm [47] is adopted here to solve the PSR-based regularization subproblem. On the other hand, $\Re_{\text{GSR}}(\mathbf{X})$ solves a group sparse coding problem for the bicubic upscaled image $\mathbf{Y}$ by forming groups of similar patches. It finds sparse representation of the ith group $\mathbf{Y}_{g_i}$ involving the dictionary $\mathbf{D}_{g_i}$ as follows:

$$\Re_{\text{GSR}}(\mathbf{X}) = \sum_i \left(\frac{1}{2} \left\| \mathbf{Y}_{g_i} - \mathbf{D}_{g_i} \boldsymbol{\beta}_{g_i} \right\|_2^2 + \lambda_2 \left\| \boldsymbol{\beta}_{g_i} \right\|_1 \right) \tag{14}$$

where $\mathbf{Y}_{g_i} = \Gamma_i(SH\mathbf{X})$ is a patch group extracted using the operator Γ from the ith search window. The above GSR regularization subproblem is efficiently solved using the accelerated split Bregman algorithm [48].

Now, to solve the composite regularization problem in (12), we adopt the variable splitting technique [49]. It is split into three relatively simpler subproblems with respect to $\boldsymbol{\alpha}$, $\boldsymbol{\beta}_{g_i}$, and $\mathbf{X}$, respectively, where the third subproblem becomes a simple least-square problem. Mathematically, these optimization problems are as follows:

$$\hat{\boldsymbol{\alpha}} = \arg\min_{\boldsymbol{\alpha}} \sum_i \left(\tfrac{1}{2} \left\| \tilde{\mathbf{y}}_i - \tilde{\mathbf{D}} \boldsymbol{\alpha}_i \right\|_2^2 + \lambda_1 \|\boldsymbol{\alpha}_i\|_1 \right) \tag{15a}$$

$$\hat{\boldsymbol{\beta}}_{g_i} = \arg\min_{\boldsymbol{\beta}_{g_i}} \sum_i \left(\tfrac{1}{2} \left\| \mathbf{Y}_{g_i} - \mathbf{D}_{g_i} \boldsymbol{\beta}_{g_i} \right\|_2^2 + \lambda_2 \left\| \boldsymbol{\beta}_{g_i} \right\|_1 \right). \tag{15b}$$

After obtaining the above two solutions, they are utilized to solve the third subproblem of finding $\mathbf{X}$ as follows:

$$\begin{aligned}\hat{\mathbf{X}} = \arg\min_{\mathbf{X}} & \tfrac{1}{2} \|\mathbf{Y} - SH(\mathbf{X})\|_2^2 \\ & + \mu_1 \sum_i \left(\tfrac{1}{2} \left\| P(SH(\mathbf{X})) - \tilde{\mathbf{D}} \alpha_i \right\|_2^2 \right) \\ & + \mu_2 \sum_i \left(\tfrac{1}{2} \|\Gamma_i(SH(\mathbf{X})) - \mathbf{D}_{g_i} \beta_{g_i}\|_2^2 \right)\end{aligned} \tag{16}$$

TABLE I
SPECIFICATIONS OF LISS-III, LISS-IV, AND PUBLICLY AVAILABLE DATASETS

Datasets	Bands	Image Size	Specifications
LISS-III	Green (Band2: 0.52-0.59 μm) Red (Band3: 0.62-0.68 μm) NIR (Band4: 0.77-0.86 μm) SWIR (Band5: 1.55-1.70 μm)	256×256	Latitude: $19°84'43''71S$ Longitude: $72°34'00''22E$ Date-of-pass: 23-Jan-2017 Spatial resolution: 23.5 m
		512×512	Latitude: $23°41'10''87S$ Longitude: $72°15'19''76E$ Date-of-pass: 18-Jan-2017 Spatial resolution: 23.5 m
LISS-IV	Green (Band2: 0.52-0.59 μm) Red (Band3: 0.62-0.68 μm) NIR (Band4: 0.77-0.86 μm)	256×256	Latitude: $25°04'03''02S$ Longitude: $66°95'44''83E$ Date-of-pass: 07-Oct-2014 Spatial resolution: 5.8 m
		512×512	Latitude: $13°44'16''52N$ Longitude: $79°58'50''52E$ Date-of-pass: 06-Feb-2017 Spatial resolution: 5.8 m
		510×510	Latitude: $25°04'03''02S$ Longitude: $66°95'44''83E$ Date-of-pass: 07-Oct-2014 Spatial resolution: 5.8 m
PatternNet	RGB images acquired via the Google Map API and Google Earth	256×256	Total classes: 38 classes Images per class: 800 Spatial resolution: 0. 062m-4.693m
UCMD	RGB images collected from USGS National Map Urban Area Imagery	256×256	Total classes: 21 classes Images per class: 100 Spatial resolution: 1 foot
AID	RGB images from Google Earth imagery	600×600	Total classes: 30 classes Images per class: 200 to 400 Spatial resolution: 0.5m to 8m

Algorithm 1: Proposed JASISR Algorithm.

Input: $\mathbf{Y}$, S, H, P, Γ, $\mathbf{D}_h$, $\mathbf{D}_\ell$, $\mathbf{D}_{g_i}$
Initialization: $t \leftarrow 0$, $\delta \leftarrow 10^{-4}$, λ_1, λ_2, μ_1, μ_2
1: **while** not converge **do**
2: $k \leftarrow k + 1$
3: $\boldsymbol{\alpha}_i^t \leftarrow$ $\arg\min_{\boldsymbol{\alpha}} \sum_i \|P(SH\mathbf{X}) - \tilde{\mathbf{D}}\boldsymbol{\alpha}_i\|_2^2 + \lambda_1 \|\boldsymbol{\alpha}_i\|_1$
4: $\boldsymbol{\beta}_{g_i}^t \leftarrow$ $\arg\min_{\boldsymbol{\beta}} \sum_i \|\Gamma_i(SH\mathbf{X}) - \mathbf{D}_{g_i}\boldsymbol{\beta}_{g_i}\|_2^2 + \lambda_2 \|\boldsymbol{\beta}_{g_i}\|_1$
5: $\mathbf{X}^t \leftarrow$ $\frac{(SH)^t Y + \mu_1 \sum_i (PSH)^t P \tilde{D} \boldsymbol{\alpha}_i + \mu_2 \sum_j (\Gamma_j SH)^t \mathbf{D}_{g_i} \boldsymbol{\beta}_{g_i}}{(SH)^t SH + \mu_1 \sum_i (PSH)^t PSH + \mu_2 \sum_i (\Gamma_i SH)^t \mathbf{D}_{g_i} \boldsymbol{\beta}_{g_i}}$
6: check convergence : $\|\mathbf{X}^t - \mathbf{X}^{t-1}\| / \|\mathbf{X}^t\| \leq \delta$
7: **end while**
Output: $\mathbf{X}^* \leftarrow \mathbf{X}^t$

where μ_1 and μ_2 are the small positive regularization parameters. Minimization problem in (16) is strictly convex and a closed-form solution can be obtained by setting its gradient w.r.t. $\mathbf{X}$ to zero. The solution is shown in step 5 of Algorithm 1.

V. EXPERIMENTS AND RESULTS

A. Dataset Preparation

In this work, the proposed SR algorithm is tested on RS images captured by ResourceSat-2 satellite and outsourced by National RS Center (NRSC),[1] Hyderabad, India, along with three standard RS datasets available in the public domain, namely, the PatternNet,[2] UC Merced Dataset (UCMD),[3] and the Aerial Image Database (AID).[4] Detailed datasets' information and experimental settings are provided in Tables I and II, while

[1]NRSC Data Center: https://uops.nrsc.gov.in/ImgeosUops/land.html
[2]PatternNet data: https://sites.google.com/view/zhouwx/dataset
[3]UCMD data: http://weegee.vision.ucmerced.edu/datasets/landuse.html
[4]AID data: https://captain-whu.github.io/AID/

TABLE II
EXPERIMENTAL SETUP AND STANDARD PERFORMANCE EVALUATION METRICS

Gaussian blur: kernel size: $K \times K$ For $2\times$: $K = 5$ and $\sigma = 1.2$ $3\times$: $K = 7$ and $\sigma = 1.6$ $4\times$: $K = 9$ and $\sigma = 2.0$
Patch ($\sqrt{m} \times \sqrt{m}$): 5×5 **Search window** ($W \times W$)**:** 20×20 **Stride:** 4
Regularization: $\lambda_1 = 0.15$, $\lambda_2 = 0.2$ **Patches in a group** (g)**:** 60
Dictionary: PSR: $\mathbf{D}_h = 25\times256$, $\mathbf{D}_\ell = 100\times256$ GSR: $\mathbf{D}_{g_i} = 25\times60$
Evaluation Metrics: **Reference-based:** PSNR, SSIM, ERGAS, SAM, Q-index, and sCC **No-reference based:** NIQE, Entropy (EN), and EME
Experimental setup: Workstation equipped with 128 GB RAM, Intel Xeon E5-2600 v4 processor, and Windows 10 OS **Tools:** MATLAB– , python 3.6.4, CUDA Toolkit 11.2, and GPU: NVIDIA Quadro 5000

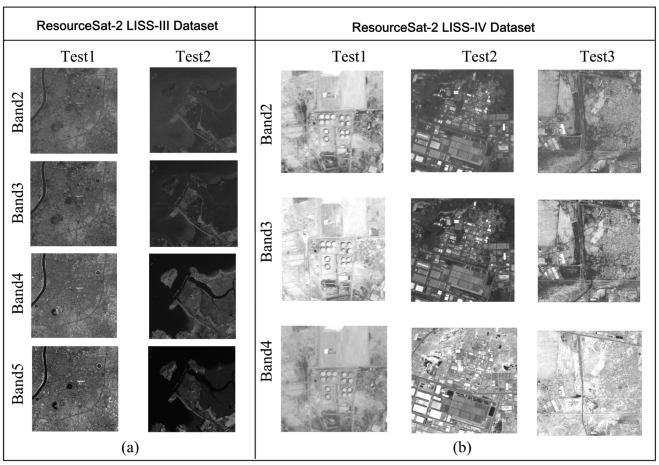


Fig. 3. Columnwise from left to right: test images of (a) LISS-III and (b) LISS-IV datasets for different bands.

a few representative images from ResourceSat-2 satellite used for performance evaluation are shown in Fig. 3.

B. Experimental Settings and Comparison to the State-of-the-Art

In this work, the LR test images for simulations are obtained by blurring and downsampling the selected images from the datasets (referred as ground truths). We have considered eight state-of-the-art patch/JSR-based SR methods, namely, the ScSR [14], Moustafa et al. [17], the CRNS [20], the GSR-SiSR [25], the RAISR [4], the CDLSR [19], the CSC-SR [18], and the LR-GSC [32] for comparisons with the proposed method. To show the competitiveness of the proposed method with the recently developed DL-based methods, we have also compared the proposed method with the SRCNN [50], the VDSR [39], the SAN [42], the CFSRCNN [40], the MHAN [41], the Beby-GAN [43], and the RCAN-it [51] for the same datasets.

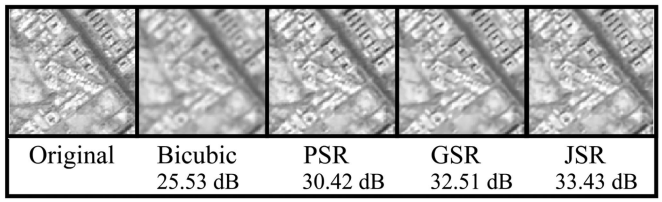


Fig. 4. Example from LISS-IV showing improved performance of JSR over PSR and GSR corresponding to a upscale ratio of 2.

C. Results

1) Ablation Test: An ablation experiment is carried out to determine the effects when we restrict either the PSR or GSR subproblem in the JASISR framework. With these settings, we obtain two variants of the proposed algorithm. Results are compared with the proposed JSR-based technique (using both PSR and GSR simultaneously), as shown in Fig. 4. It can be seen that when JSR is not used, contrast produced by PSR or GSR alone is not significantly different. There are also drops in PSNR values, about 3 dB in the case of PSR and roughly 1 dB in the case of GSR. Furthermore, it is also clear from the experimental results of Fig. 5 and Table III that methods based on only PSR and GSR (e.g., in ScSR or in GSRGSiSR) fail to produce significant improvements. Apart from lacking in quantitative metrics, method based on only PSR, such as ScSR, is unable to adequately decrease the serrated edges or ringing effects. However, GSRGSiSR attempts to oversmooth the images for higher upscale ratios. In this scenario, the proposed JSR method produces structurally sharpened and contrast improved high PSNR images.

2) Visual Interpretation: In the proposed work, we carry out SR of each band image from the MS image separately. We perform $2\times$ and $4\times$ upscaling of two test images, namely, Test1 and Test2, while for another test image, Test3 (results provided in the supplementary file), we choose to perform $2\times$ and $3\times$ SR from respective undersampled LR images. The reconstructed band images of the LISS-IV dataset for different methods are shown in Fig. 5 for all the test images. From visual inspection, it is clear that the proposed method provides the best perceptual quality of reconstructed images. Furthermore, error images between original and reconstructed images of Test1 and Test2 are also shown in Fig. 6 for better interpretation. The results demonstrate that the least error is obtained in case of the proposed method.

The proposed method is also tested using some of the public domain RGB RS datasets and results are compared in Fig. 7. It is clearly able to maintain better performance than others both visually and in terms of PSNR and structural similarity (SSIM).

3) Quantitative Analysis: Ideally, for a better reconstructed image with richer information, values of PSNR, SSIM, Q-index, enhanced measure evaluation (EME), and EN should increase, while that of erreur relative globale adimensionnelle de synthese (ERGAS), spectral angle mapper (SAM), and natural image

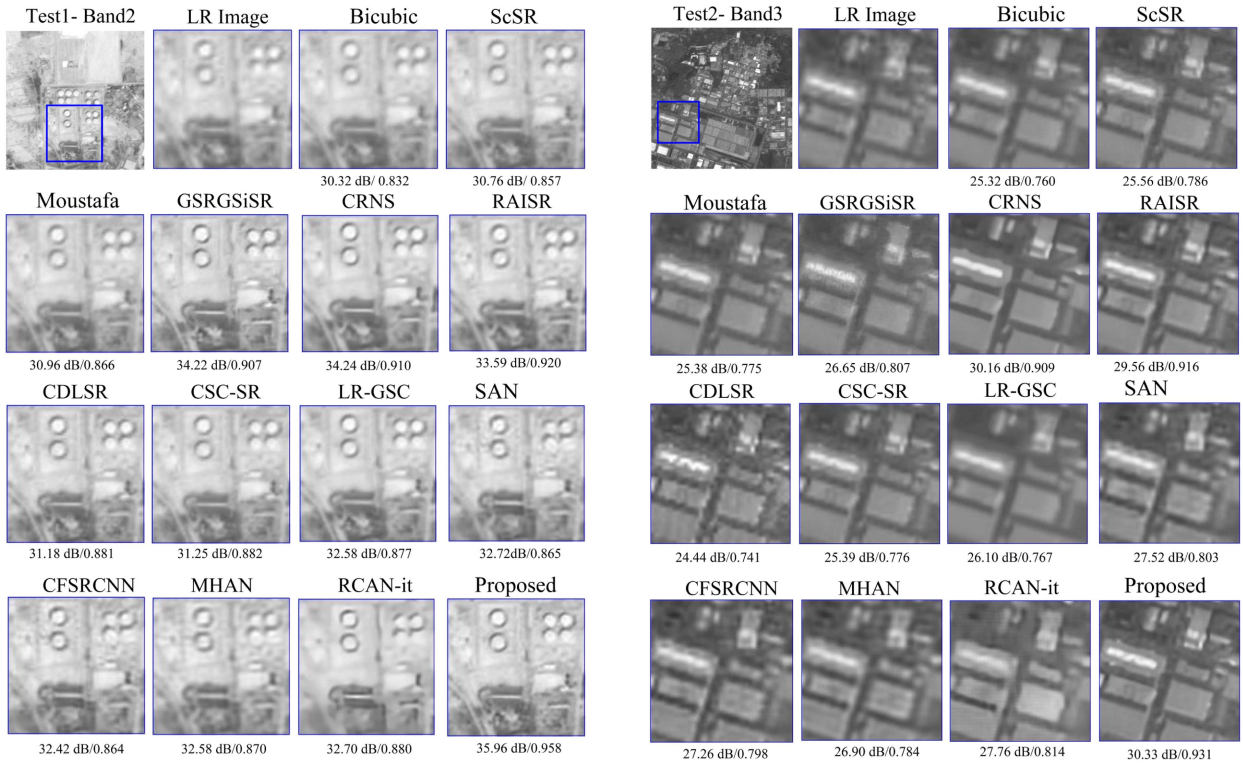


Fig. 5. Visual comparisons of the SR outputs for LISS-IV dataset. (a) Rows 1–3 "Test1" band2. (b) Rows 1–3 "Test2" band3 images for upscale ratios of 2 and 4 using different sparse representation-based methods. The PSNR and SSIM measures for all the visual outputs are provided at the bottom of image for better interpretation.

TABLE III
AVERAGE QUANTITATIVE MEASURES ACROSS ALL THE BANDS FOR THE LISS-III DATASET

	Parameters	Bicubic	ScSR	Moustafa	GSRGSiSR	CRNS	RAISR	CDLSR	CSC-SR	LR-GSC	Proposed
Test1 (zoom = 2)	PSNR (dB)	33.91	34.27	34.36	37.10	36.74	36.29	35.97	34.09	36.21	**37.54**
	SSIM	0.8689	0.8789	0.8833	0.9111	0.9312	9192	0.9060	0.8715	0.9149	**0.9366**
	sCC	0.9549	0.9572	0.9594	0.9776	0.9781	9735	0.9712	0.9562	0.9730	**0.9797**
Test2 (zoom = 3)	PSNR (dB)	24.12	24.27	24.32	24.87	25.42	25.55	24.58	24.28	24.94	**25.90**
	SSIM	0.3904	0.4033	0.4063	0.4372	0.4748	0.5124	0.4267	0.4183	0.4262	**0.5745**
	sCC	0.7297	0.7320	0.7330	0.7689	0.7997	0.8083	0.7699	0.7305	0.7776	**0.8227**

Best results are indicated by boldface.

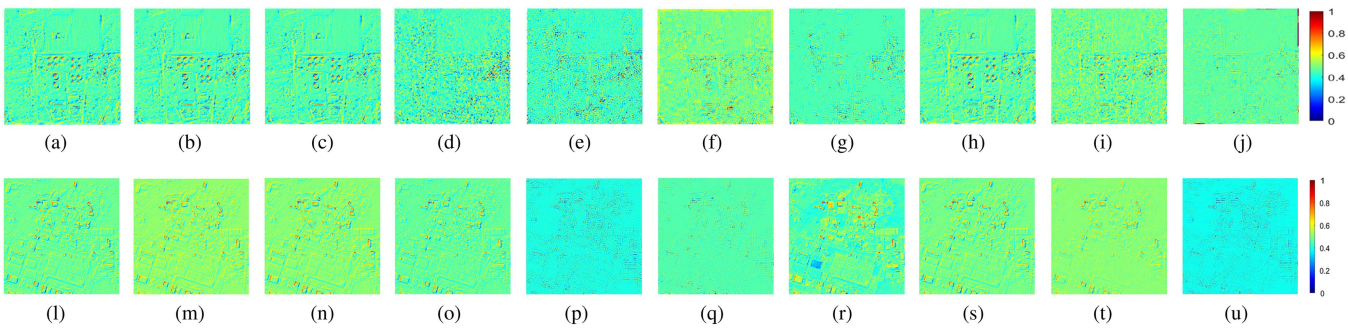


Fig. 6. Error between original and reconstructed images of LISS-IV: first row - Test1-band2 with upscale ratio 2; second row - Test2-band3 with upscale ratio 4.

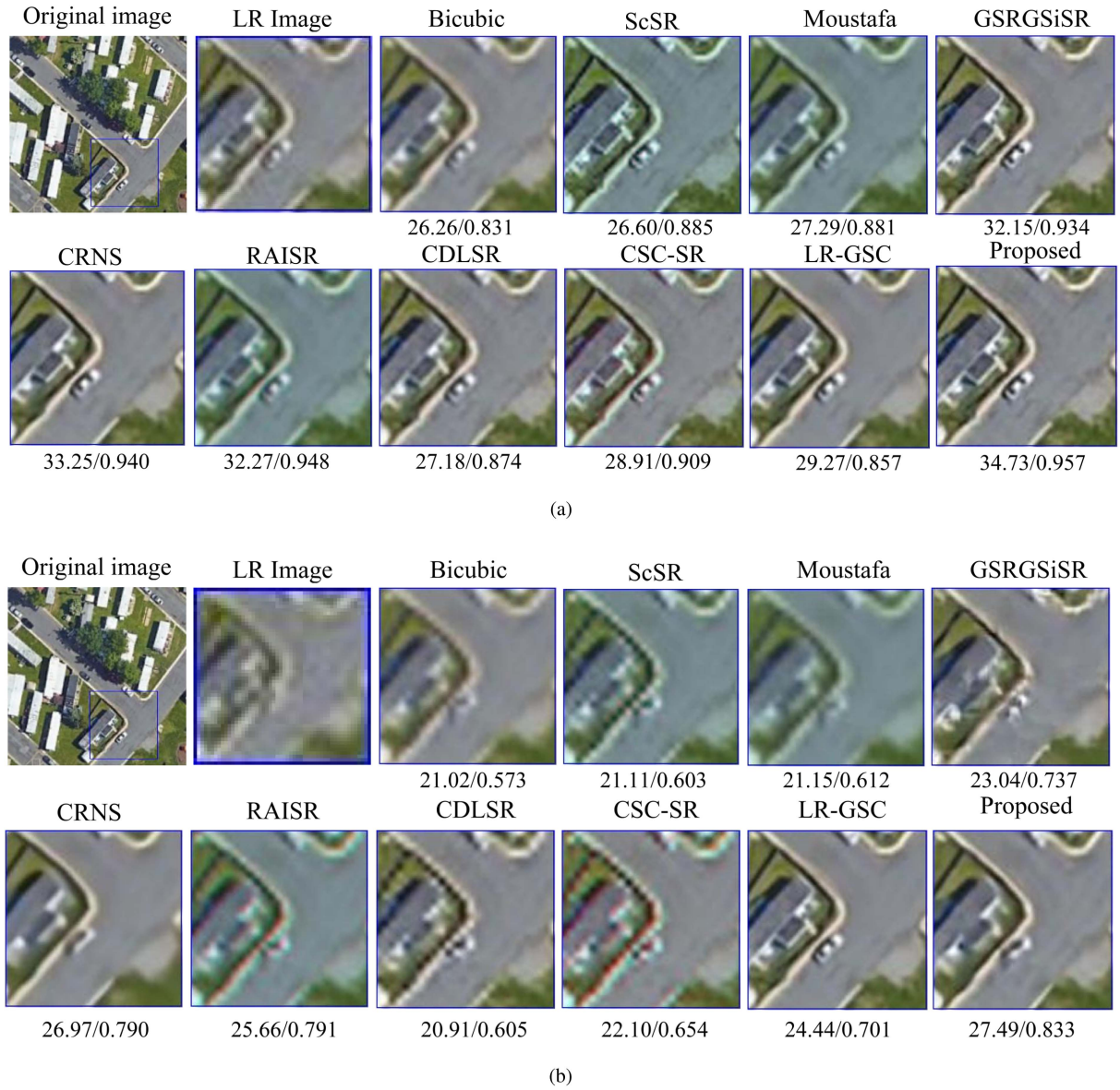


Fig. 7. Visual comparison of SR results for different methods on a test image of PatternNet for zooming factors of 2 and 4. The PSNR and SSIM measures for all the visual outputs are provided at the bottom of each image for better interpretation. (a) Upscale ratio 2. (b) Upscale ratio 4.

quality evaluator (NIQE) should decrease compared with others. We evaluate different methods based on eight quantitative parameters mentioned above and the results are shown in Table III and Fig. 8 for Test1 with 2× upscaling and Test2 with 3× upscaling, respectively, for the LISS-III dataset. We observe that average PSNR is the highest in case of the proposed method that is 37.54 dB for Test1 and 24.90 dB for Test2; resulting on an average 0.5–3.63 dB improvements over other techniques. GSRGSiSR and CRNS are the closest in terms of PSNR and SSIM with the proposed. However, the proposed method clearly outperforms GSRGSiSR and CRNS with improvements of 0.26–0.82, 0.29–0.39, and 11.20–12.92 in terms of ERGAS, SAM, and NIQE, respectively, in case of Test1 for 2× upscaling. Other methods, such as ScSR, CSC-SR, and LR-GSC, result in less PSNR with differences of 3.27 dB, 3.45 dB, and 1.33 dB, respectively. Similarly, SSIM is also the highest in case of the proposed method with average improvement of 0.005–0.18 over others, while ERGAS and SAM values are the least in most cases. Spatial correlation coefficient (sCC) value is slightly higher in case of CRNS and RAISR both for 2× and 3× upscaling, while the proposed method gives the best EME value in all cases. It should also be noted that with learned dictionaries of sizes greater than 256, PSNR slightly improves for all the test images, but at the cost of considerable computational time.

4) Comparison With DL: We compare the performance of the proposed method with seven state-of-the-art DL-based SR methods, as mentioned in Section V-B. Simulations are carried out using three publicly available RS datasets. Since these images are in RGB format, we first transform them into YCbCr format and apply the proposed algorithm on the *Y*-channel.

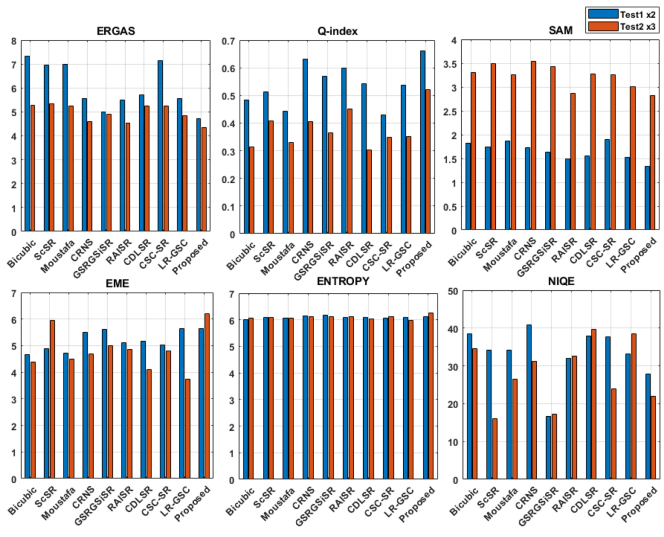


Fig. 8. Comparison of ERGAS, SAM, Q-index, EME, Entropy, and NIQE values for Test1 and Test2 images of LISS-III corresponding to upscale ratios 2 and 3, respectively.

We have selected 80%–90% images of each datasets for training the network and 10% is used for validation. We select a few test images from the above datasets randomly such that they are not a part of the training. To provide a fair comparison with the non-DL methods, LR test images for evaluating the DL-based SR methods are generated by the same procedure. While preparing for the LISS-IV training datasets, equal number of RoI images of size 256×256 need to be selected from the original images (of size, say, $10\,000 \times 10\,000$). RoI images selected from individual bands are based on their entropy and variance values. The SR reconstruction is carried out for 7–8 test images from the PatternNet, UCMD, and AID datasets using the seven DL methods. Visual outputs of three test images, namely, Airplane, Buildings, and Bridge from these datasets, are shown in Fig. 9, while average PSNR and SSIM comparisons are shown in Table IV, for upscale ratios 2, 3, and 4, respectively. In visual inspection, it is observed that high-frequency information is not well preserved by the DL-based methods from blurred images. On the other hand, the reconstructed images of the proposed method have more fine details and edges compared with others. Similarly, from Table IV, it is observed that the average PSNR and SSIM of the proposed method are higher compared with the DL-based SR methods for all the test images. The improvement in PSNR is in the range of 2–3 dB, while that of SSIM ranges in 0.1–0.15, approximately. Furthermore, the last row of Fig. 5 shows that the proposed method performs better than the DL-based methods both visually and in terms of PSNR/SSIM for the LISS-IV dataset as well.

In terms of computational complexity, the DL-based methods take $\approx$ 5–8 h for network training, and 10–100 s for the reconstruction of a 256×256 image. However, the proposed method takes only $\approx$ 350–400 s for learning adaptive dictionaries as well as reconstruction of an HR image of the same size.

5) Convergence Test: An empirical test is done by plotting PSNR values of the reconstructed images obtained with varying number of iterations of the proposed algorithm. Fig. 10(a) and (b) shows the convergence plot of all the three bands for Test1 and Test2 images, respectively. We consider 30 iterations to reduce the computation time as it is observed that PSNR increases gradually up to 20–30 iterations and beyond that it becomes almost flat.

6) Computational Complexity and CPU Time Comparison: Major computation comes from two aspects—first, learning the K-SVD dictionary and solving the PSR problem, and second, solving the GSR problem. If K is the number of atoms in the dictionary, then PSR needs $\mathcal{O}(K^3)$ operations. If N is the total number of pixels, t_s is the time required for searching similar patches for any given patch, and the complexity of SVD dictionary learning from each group is $\mathcal{O}(m \times k)$. Therefore, the overall complexity of the GSR part can be given as $\mathcal{O}(N(m \times k + t_s))$. The overall computational cost of the JSR problem is $\mathcal{O}(K^3 + N(m \times k + t_s))$. The execution time of the proposed method is graphically compared with those of the other methods in Fig. 11. In this work, since we have performed bandwise SR, we compute the execution time for the reconstruction of a test image as the average time over all of its bands. Here, the first plot (top) compares the average SR reconstruction time taken by different methods for equal number of test images randomly taken from each of the dataset. Also, to get an idea of the underlying experimentation on different test images, we have shown a comparison of average execution times against the three test images of LISS-IV for different methods (bottom part).

7) Parallel Implementation Using GPGPU Hardware: We have used GPGPU hardware to accelerate the execution time of the proposed JSR-based MS image SR using the CUDA programming model in the MATLAB parallel computing environment. The major time consuming parts in the sequential program are the ℓ_1-feature sign for sparse representation (PSR) and GSR (block matching, group-sparse coding via SVD). A CPU–GPU hybrid environment is utilized to speed up the code executions in this work. Here, the PSR is implemented using CUDA-mex and the other functions are implemented using C++ mex functions in the MATLAB parallel computing environment.

In sequential version of the proposed JASISR, the PSR and GSR subproblems in CPU require 4.82 s and 5.82 s, respectively, which are called in an inner loop, which in turn is called in an outer loop of the ADMM algorithm. The time taken by PSR and GSR for each patch using parallel implementation is approx. 0.70 s and 0.36 s, respectively. Hence, the overall execution time of the algorithm is accelerated by several times. We have compared the run time of the CPU versus GPU implementations of the proposed algorithm for different images sizes starting from $128 \times 128 \times 4$ to $1024 \times 1024 \times 3$, as shown in Table V. Here, we have additionally used the MATLAB "parfor" instruction to simultaneously process all the bands of the MS images, where each image will be reconstructed through the proposed MEX-CUDA-based parallelization.

8) Parameter Sensitivity Analysis: Selection of different regularization parameters is examined by conducting a few parameter sensitivity experiments. For this, the PSNR values obtained

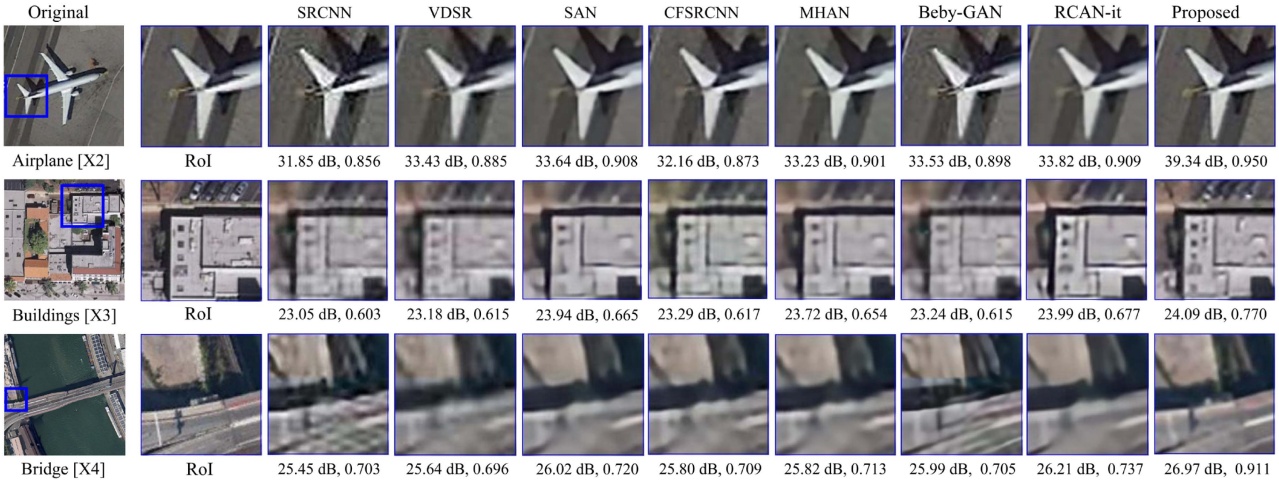


Fig. 9. Comparison of reconstructed images of DL-based methods with the proposed method for different datasets. From top to bottom: first row - PatternNet (upscale ratio: 2), second row- UC Merced (upscale ratio: 3), and the last row for AID images (upscale ratio: 4). From left to right: original, RoI, reconstructed images using different methods: SRCNN, VDSR, SAN, CFSRCNN, MHAN, Beby-GAN, RCAN-it, and the proposed method.

TABLE IV
PERFORMANCE EVALUATION OF DL AND THE PROPOSED METHOD FOR LAND COVER DATASETS

	Test Images	SRCNN [38]			VDSR [39]			SAN [42]			CFSRCNN [40]			MHAN [41]			BeBy-GAN [43]			RCAN-it [51]			Proposed		
		X2	X3	X4	X2	X3	X4	X2	X3	X4	X2	X3	X4	X2	X3	X4	X2	X3	X4	X2	X3	X4	X2	X3	X4
		PSNR/ SSIM	PSNR/ SSIM	PSNR/ SSIM	PSNR/ SSIM	PSNR/ SSIM	PSNR/ SSIM	PSNR/ SSIM	PSNR/ SSIM	PSNR/ SSIM	PSNR/ SSIM	PSNR/ SSIM	PSNR/ SSIM	PSNR/ SSIM	PSNR/ SSIM	PSNR/ SSIM	PSNR/ SSIM	PSNR/ SSIM	PSNR/ SSIM	PSNR/ SSIM	PSNR/ SSIM	PSNR/ SSIM	PSNR/ SSIM	PSNR/ SSIM	PSNR/ SSIM
PatternNet	Airplane	31.85 0.856	30.07 0.796	28.01 0.738	33.43 0.885	30.42 0.821	28.51 0.770	33.64 0.908	31.31 0.856	29.33 0.810	32.16 0.873	30.04 0.815	29.29 0.791	33.23 0.901	30.65 0.844	28.96 0.797	33.53 0.898	30.24 0.813	28.77 0.782	33.82 0.909	31.69 0.867	29.41 0.828	**39.34 0.950**	**34.07 0.859**	**30.83 0.881**
	Cemetery	27.06 0.611	26.09 0.549	25.32 0.481	27.94 0.686	**26.31** 0.555	**25.57** 0.493	27.62 0.674	26.08 0.547	25.51 0.494	27.31 0.658	25.75 0.513	25.47 0.484	27.55 0.670	26.05 0.541	25.45 0.491	27.58 0.642	26.21 0.535	25.48 0.488	27.84 0.681	26.21 0.550	25.54 0.502	**28.36 0.894**	24.92 **0.765**	23.45 **0.595**
	Ferry terminal	25.45 0.776	23.58 0.683	21.73 0.603	25.71 0.813	23.12 0.699	21.64 0.623	25.84 0.826	23.64 0.744	21.95 0.672	25.46 0.807	22.95 0.708	21.99 0.663	25.67 0.821	23.39 0.732	21.81 0.664	25.74 0.804	23.49 0.699	21.73 0.641	26.04 0.832	23.94 0.753	22.01 0.689	**29.05 0.907**	**24.63 0.790**	**22.22 0.764**
	Intersection	26.55 0.697	25.34 0.602	23.97 0.507	27.21 0.708	25.56 0.599	**24.43** 0.514	27.12 0.712	25.58 0.609	24.38 0.521	26.81 0.693	25.04 0.559	24.36 0.507	27.07 0.708	25.51 0.599	24.32 0.511	27.06 0.701	25.39 0.584	24.37 0.511	27.27 0.716	25.69 0.608	24.42 0.536	**28.31 0.788**	**25.72 0.641**	24.26 **0.621**
	Mobilehomepark	27.46 0.793	25.51 0.696	23.34 0.615	29.28 0.855	25.91 0.732	23.97 0.633	29.07 0.863	26.24 0.762	24.47 0.681	28.11 0.836	24.98 0.716	24.48 0.663	28.82 0.859	25.97 0.754	24.12 0.662	28.49 0.835	25.73 0.721	24.28 0.638	29.25 0.867	26.59 0.773	24.55 0.701	**34.28 0.963**	**29.70 0.907**	**26.17 0.815**
	Nursing home	24.84 0.681	23.47 0.599	21.91 0.489	25.41 0.721	23.52 0.606	22.34 0.517	25.64 0.737	23.72 0.645	22.52 0.568	25.17 0.712	23.07 0.593	**22.53** 0.544	25.46 0.731	23.55 0.631	22.29 0.547	25.41 0.711	23.44 0.596	22.43 0.521	25.63 0.736	23.87 0.638	22.36 0.581	**27.29 0.848**	**24.14 0.734**	22.34 **0.657**
	Wastewaterplant	30.49 0.811	28.67 0.730	26.55 0.638	32.38 0.863	29.17 0.751	27.31 0.671	32.57 0.878	29.34 0.784	27.38 0.706	31.51 0.855	28.29 0.731	27.24 0.688	32.37 0.876	29.06 0.775	27.23 0.699	32.49 0.870	29.05 0.745	27.31 0.671	32.54 0.879	29.87 0.797	**27.40** 0.726	**36.55 0.958**	**32.97 0.879**	26.55 **0.829**
	Average	27.67 0.747	26.10 0.665	24.40 0.581	28.76 0.790	26.28 0.680	24.82 0.603	28.78 0.799	26.55 0.706	25.07 0.636	28.07 0.776	25.73 0.662	25.05 0.620	28.60 0.795	26.31 0.696	24.88 0.624	28.61 0.780	26.22 0.670	24.91 0.607	28.91 0.802	26.83 0.712	25.09 0.651	**31.88 0.901**	**28.02 0.796**	**25.11 0.737**
UC Merced	Airplane	27.37 0.778	26.22 0.715	24.73 0.651	28.55 0.810	26.81 0.736	25.51 0.679	29.49 0.841	27.71 0.770	**26.57** 0.726	28.04 0.774	26.55 0.710	26.44 0.706	29.27 0.833	27.41 0.765	26.16 0.718	28.95 0.795	26.90 0.714	25.95 0.676	29.82 0.848	27.78 0.752	26.55 0.734	**32.33 0.952**	**29.04 0.913**	26.18 **0.793**
	Building	24.74 0.722	23.05 0.603	21.52 0.491	25.01 0.735	23.18 0.615	21.96 0.518	26.33 0.796	23.94 0.665	**22.39** 0.568	25.19 0.736	23.29 0.617	22.35 0.551	26.04 0.784	23.72 0.654	22.26 0.556	25.56 0.782	23.24 0.615	22.31 0.538	26.18 0.815	23.99 0.677	22.38 0.588	**28.61 0.905**	**24.09 0.770**	22.21 **0.648**
	Denseresidential	25.01 0.726	23.69 0.639	22.46 0.562	25.82 0.765	24.23 0.671	23.05 0.594	26.45 0.801	24.64 0.711	23.36 0.641	25.42 0.735	24.07 0.661	23.38 0.627	26.26 0.792	24.46 0.703	23.25 0.635	25.93 0.757	24.05 0.660	23.35 0.598	26.69 0.822	24.52 0.699	23.67 0.663	**29.28 0.946**	**26.30 0.899**	**24.16 0.758**
	Mediumresidential	28.43 0.821	26.43 0.719	24.73 0.622	29.63 0.847	27.30 0.752	25.53 0.657	30.33 0.875	27.59 0.774	25.97 0.693	29.23 0.835	26.87 0.727	26.15 0.684	30.13 0.869	27.42 0.768	25.78 0.684	29.78 0.843	27.46 0.737	25.77 0.667	30.41 0.884	27.87 0.776	26.08 0.710	**34.95 0.962**	**29.77 0.894**	**27.05 0.797**
	Mobilehomepark	20.15 0.601	18.62 0.447	17.27 0.317	19.93 0.603	18.42 0.444	17.37 0.326	21.18 0.681	**19.21** 0.518	17.99 0.404	20.19 0.614	18.80 0.461	17.96 0.397	20.89 0.665	19.04 0.502	17.86 0.393	20.76 0.659	18.74 0.472	17.88 0.412	21.39 0.724	19.37 0.545	18.09 0.438	**22.49 0.833**	19.10 **0.646**	**18.11 0.491**
	Overpass	27.41 0.721	25.71 0.641	24.27 0.544	28.11 0.762	26.21 0.654	24.79 0.564	28.78 0.781	**26.98** 0.681	**25.68** 0.617	28.19 0.763	26.05 0.636	25.41 0.581	28.63 0.776	26.79 0.676	25.15 0.583	28.42 0.752	26.94 0.677	25.40 0.588	27.94 0.786	26.49 0.682	25.35 0.621	**30.81 0.868**	26.93 **0.724**	24.86 **0.652**
	River	23.54 0.655	22.14 0.521	21.23 0.442	23.82 0.667	22.43 0.544	21.58 0.465	23.98 0.687	22.39 0.549	21.51 0.472	23.54 0.658	22.05 0.515	**21.59** 0.475	23.89 0.681	22.34 0.547	21.45 0.467	23.85 0.680	22.48 0.543	21.61 0.469	24.06 0.732	22.44 0.568	21.50 0.498	**25.93 0.838**	**22.76 0.654**	21.34 **0.592**
	Average	25.23 0.717	23.69 0.612	22.31 0.518	25.83 0.741	24.08 0.630	22.82 0.543	26.64 0.780	24.63 0.666	23.35 0.588	25.68 0.730	23.95 0.618	23.32 0.574	26.44 0.771	24.45 0.659	23.13 0.576	26.17 0.752	24.25 0.631	23.18 0.564	26.64 0.801	24.63 0.671	23.37 0.607	**29.20 0.900**	**25.42 0.785**	**23.41 0.675**
AID	Baseballfield	29.72 0.822	28.09 0.754	26.39 0.689	30.34 0.832	28.22 0.757	26.73 0.696	30.92 0.854	28.48 0.774	27.09 0.718	30.31 0.831	28.27 0.761	26.93 0.706	30.63 0.846	28.32 0.767	26.94 0.710	30.54 0.852	28.36 0.758	27.07 0.706	31.22 0.858	28.81 0.779	27.20 0.727	**34.83 0.972**	**29.77 0.931**	**27.51 0.902**
	Bridge	28.33 0.801	26.98 0.752	25.45 0.703	28.99 0.811	27.17 0.753	25.64 0.696	29.33 0.827	27.33 0.767	26.02 0.720	28.56 0.803	27.07 0.755	25.80 0.709	28.97 0.819	27.05 0.760	25.82 0.713	29.12 0.820	27.26 0.752	25.99 0.705	29.62 0.833	27.67 0.772	26.21 0.737	**31.71 0.930**	**28.85 0.877**	**26.97 0.911**
	Church	28.15 0.796	26.21 0.710	24.23 0.635	28.79 0.809	26.35 0.718	24.42 0.655	29.44 0.865	26.84 0.780	25.16 0.707	28.69 0.828	26.54 0.756	24.90 0.685	29.14 0.855	26.62 0.768	24.88 0.691	29.01 0.815	26.48 0.734	25.05 0.666	29.69 0.870	27.09 0.785	25.25 0.722	**32.85 0.967**	**28.35 0.915**	**25.81 0.891**
	Commercial	24.75 0.773	22.84 0.654	20.69 0.528	25.51 0.785	23.07 0.654	21.03 0.497	25.71 0.777	23.33 0.657	21.72 0.550	25.12 0.741	23.05 0.626	21.44 0.526	25.45 0.770	23.06 0.637	21.43 0.529	25.67 0.790	23.19 0.638	21.57 0.526	25.79 0.782	23.43 0.652	21.63 0.564	**28.42 0.900**	**24.28 0.868**	**21.83 0.773**
	Industrial	28.67 0.816	26.87 0.730	24.89 0.637	29.43 0.831	27.01 0.729	25.37 0.624	30.01 0.850	27.20 0.730	25.49 0.638	29.45 0.825	26.93 0.714	25.32 0.629	29.89 0.843	27.06 0.722	25.38 0.633	29.55 0.841	27.12 0.721	25.44 0.630	30.46 0.856	27.62 0.743	25.53 0.664	**35.21 0.965**	**30.08 0.912**	**27.05 0.893**
	Mediumresidential	24.98 0.685	23.39 0.544	21.96 0.428	25.49 0.691	23.49 0.536	22.09 0.408	25.69 0.682	23.63 0.533	**22.53** 0.446	25.15 0.648	23.47 0.514	22.39 0.428	25.41 0.668	23.46 0.520	22.34 0.432	25.65 0.701	23.62 0.524	22.52 0.435	25.93 0.690	23.86 0.543	22.44 0.459	**28.21 0.893**	**24.11 0.874**	22.23 **0.763**
	Resort	24.51 0.699	23.32 0.595	22.11 0.509	24.88 0.706	23.35 0.590	22.25 0.489	24.95 0.706	23.42 0.595	22.46 0.518	24.62 0.678	23.32 0.582	22.42 0.509	24.77 0.692	23.31 0.586	22.38 0.510	24.99 0.719	23.43 0.580	22.38 0.470	25.17 0.709	23.62 0.597	**22.54** 0.528	**26.23 0.880**	**23.47 0.910**	22.07 **0.837**
	School	28.76 0.816	26.83 0.736	24.55 0.637	29.39 0.826	27.01 0.737	24.94 0.643	30.11 0.865	27.30 0.762	**25.52** 0.675	29.74 0.842	27.16 0.750	25.32 0.664	30.02 0.858	27.23 0.757	25.37 0.668	29.84 0.832	27.21 0.745	25.64 0.665	30.47 0.870	27.67 0.771	24.20 0.610	**35.31 0.957**	**29.75 0.963**	22.62 **0.913**
	Average	27.23 0.775	25.69 0.684	23.78 0.595	27.85 0.786	25.70 0.684	24.06 0.588	28.27 0.803	25.94 0.699	24.49 0.621	29.70 0.774	25.72 0.682	24.31 0.607	28.03 0.793	25.76 0.689	24.31 0.610	28.04 0.796	25.83 0.682	24.45 0.600	28.54 0.808	26.22 0.705	24.38 0.626	**31.22 0.933**	**27.33 0.906**	**24.51 0.860**

TABLE V
COMPARISON OF SEQUENTIAL AND PARALLEL IMPLEMENTATION TIME FOR DIFFERENT IMAGES

Input size	Output size	Seq. time	Parallel time
128×128×4	256×256×4	1365 s	57 s
256×256×3	512×512×3	3443 s	261 s
512×512×4	1024×1024×4	26463 s	787 s
1024×1024×3	2048×2048×3	44478 s	1178 s

by the proposed algorithm are recorded by varying a single parameter while keeping the remaining parameters fixed. Fig. 12(a) shows the sensitivity of λ for dictionary training; the PSNR reaches the maximum at $\lambda = 0.15$ when the latter is varied in the range of 0.11–0.19. Similarly, Fig. 12(b) and (c) shows the effects λ_1 of PSR and λ_2 of GSR, respectively. From the plots, it is found that the best values of λ_1 and λ_2 are 0.15 and 0.2, respectively. In Fig. 12(d), we have shown the tradeoff

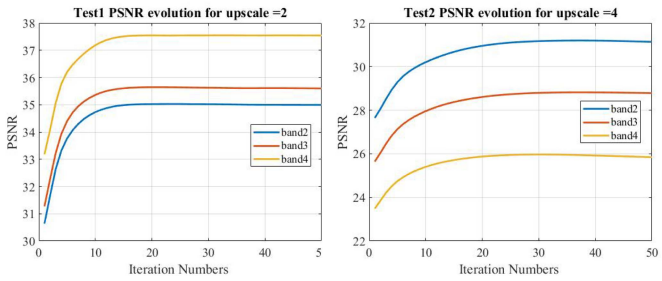


Fig. 10. Convergence analysis of the proposed method shown in terms of progression of PSNR values with respect to the iteration number. (a) For "Test1" image with upscale ratio 2. (b) For "Test2" image with upscale ratio 4 of LISS-IV dataset.

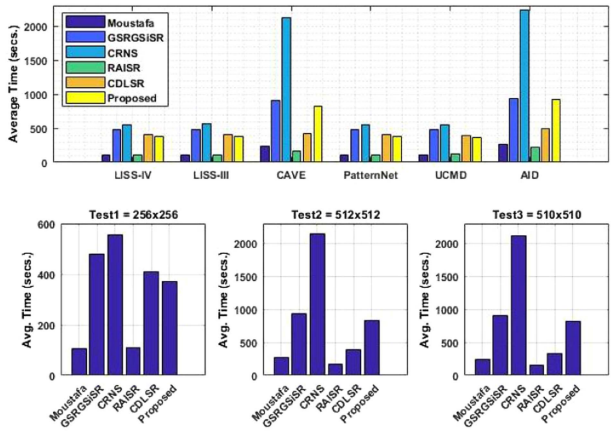


Fig. 11. Average reconstruction time of different methods (corresponding to upscale ratio 2) for equal number of randomly selected images from different datasets (top), and average bandwise reconstruction time of different methods for the three test images of LISS-IV dataset.

between the parameters μ_1 and μ_2 by taking a range of values for $\mu_1 \in [1 \times 10^{-5}, 7 \times 10^{-5}]$ and $\mu_2 \in [1 \times 10^{-3}, 7 \times 10^{-3}]$, while the best combination is found as $\mu_1 = 0.00005$ and $\mu_2 = 0.005$.

9) SR of Real LR Images: Experiments are also carried out to check the performance of the proposed algorithm on real RS images. Unlike the previous experiments, here, we input the LR test image directly to the SR algorithm, without any preprocessing, i.e., without blurring and downsampling. In Fig. 13, we have shown comparisons of SR results for 2× and 4× upscalings of a LISS-IV test image of size 128×128. Since we do not have the ground-truth image, we compute a few no-reference-based quantitative metrics, such as NIQE, EN, and EME (as mentioned in Table II) for performance evaluation. Ideally, lower NIQE and greater EN and EME values indicate better perceptual quality. It can be seen that the proposed method produces a sharper image having fine details. It also yields the lowest NIQE scores (17.34 for 2× and 35.62 for 4× upscalings) indicating the best naturalness among all the outputs. Similarly, we also achieve the highest scores for EN and EME for 2× (EN=6.857 and EME=3.141) and 4× upscalings (EN=35.62 and EME=7.219), respectively.

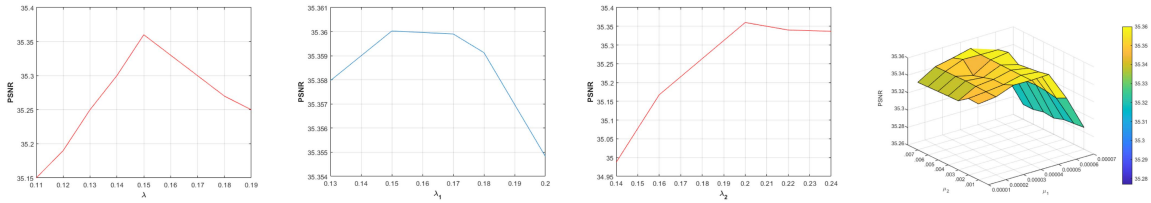
10) Land Cover Classification: We have performed classification on the HR reconstructed images obtained from different methods to interpret the image contents and analyze the effects of SR algorithm on a given LR MS image. ENVI classic 5.1 is used for classification and analyzing the results. We have conducted the experiment with two test images having different number of classes. In the first test image, regions of interests (ROIs) are labeled as bare land (green), buildings (blue), and vegetation (red), while in the second image, RoIs are labeled as road (red), green field (green), mobile home (blue), and vegetation (yellow). Multiple RoIs are selected under each class for training with the test images. Fig. 14(a) and (b) shows the classification results obtained for the two test images from LISS-IV and PatternNet datasets, respectively.

It is significantly noticeable that the proposed method has the most similar classified regions with the original image. An average accuracy of pixel counts is also computed across different classes for both the images. It is observed that the maximum accuracy is obtained in case of the proposed method, while in case of the ScSR, the pixel count is quite different from both the original and LR images. We also calculate the land cover area under each classified region from their pixel counts and standard per-pixel resolution of the sensor. The average pixel count accuracy of LR image (interpolated to original size) with respect to the original is about 88.55% and that of the proposed method is more than 95.96% for the PatternNet image. While for the LISS-IV image, they are 98.37% and 99.85% for the LR image and the proposed method, respectively.

VI. Discussion

The proposed JSR-based method is able to restore the texture as well as structure components better than others. Although some of the sparse-based methods, such as the CRNS, the GSRGSiSR, the LR-GSC, and the RAISR, are slightly closer in terms of PSNR, however, smaller details in their reconstructed images tend to be smoothed. GSRGSiSR lacks in recovering the edges and sharp details at higher upscalings. Moreover, when compared with other RS evaluation metrics, such as ERGAS, SAM, and NIQE, the proposed method performs perceptually better than these methods. The results also demonstrate that the proposed method produces the least error compared with others. It is also faster than its closest competitors: GSRGSiSR, CRNS, LR-GSC, and CDLSR. However, RAISR outperforms the proposed method in terms of execution time, but its reconstruction quality is poor. This is because the objective of the former is to achieve rapid reconstruction using a simple mapping from LR-to-HR without targeting high accuracy in SR reconstruction. We have not compared the proposed method with the ScSR for reconstruction time as the dictionary training in the latter is done separately from an external HR dataset.

In comparison with some of the state-of-the-art DL-based methods, it is observed that fine details and edges are restored well by the proposed method from the blurred and downsampled images. However, this observation is somewhat surprising for the authors, the reason may be probably due to the blurring imposed by degradation model on the images before training



(a) (b) (c) (d)

Fig. 12. Parameter sensitivity analysis of the proposed algorithm. (a) PSNR versus λ of dictionary. (b) PSNR versus λ_1 of PSR. (c) PSNR versus λ_2 of GSR. (d) Tradeoff between μ_1 and μ_2.

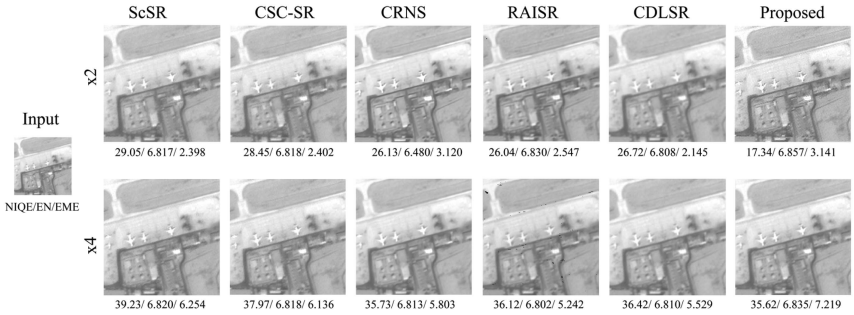
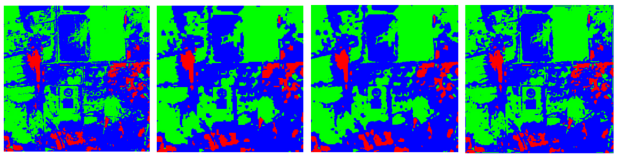


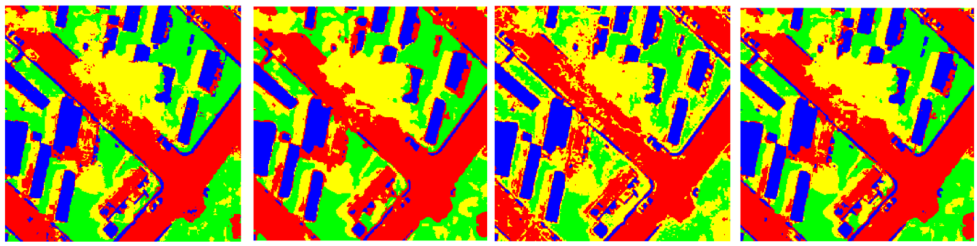
Fig. 13. Comparison of real image SR outputs by the proposed method and other SR methods.



Classes	Original		LR Image			ScSR			Proposed		
	Pixel Counts	Area (Sq. Km)	Pixel Counts	Area (Sq. Km)	Pixel Count Acc. (%)	Pixel Counts	Area (Sq. Km)	Pixel Count Acc. (%)	Pixel Counts	Area (Sq. Km)	Pixel Count Acc. (%)
Bare Land	27,243	0.9164	26,916	0.9054	98.79	27,731	0.9328	98.20	27,276	0.9175	99.87
Buildings	33,689	1.1332	33,872	1.1394	99.45	34,219	1.1511	98.42	33,639	1.1316	99.85
Vegetation	4,604	0.1548	4,748	0.1596	96.87	4,586	0.1542	99.60	4,611	0.1550	99.84
Average Pixel Count Accuracy:					98.37%			98.74%			**99.85%**

* Pixel Count Accuracy = [1-{abs(org- rec)/org}]*100

(a)



Classes	Original		LR Image			ScSR			Proposed		
	Pixel Counts	Area (Sq. mtr)	Pixel Counts	Area (Sq. mtr)	Pixel Count Acc. (%)	Pixel Counts	Area (Sq. mtr)	Pixel Count Acc. (%)	Pixel Counts	Area (Sq. mtr)	Pixel Count Acc. (%)
Road	21,541	1324	25,377	1560	84.88	21,526	1323	99.93	22,507	1384	95.70
Green Field	17,028	1048	15,784	972	92.69	11,758	723	69.05	16,051	986	94.26
Mobile Home	12,059	742	9,343	574	77.47	11,007	677	91.27	11,652	717	96.62
Trees	14,908	916	15,032	924	99.17	21,245	1307	70.17	15,326	943	97.27
Average Pixel Count Accuracy:					88.55%			82.61%			**95.96%**

* Pixel Count Accuracy = [1- {abs(org -rec)/org}]*100

(b)

Fig. 14. Results of supervised classification performed on (a) LISS-IV "Test1" image, and (b) PatternNet "Mobilehomepark" image. From left to right: original image, LR image, and reconstructed images of ScSR and the proposed method.

by the DL networks. It will be interesting to see future developments of DL models that can counter the image degradation component in a more systematic manner. Moreover, in terms of computational complexity, the DL methods take several hours for network training compared with overcomplete dictionary training in shallow learning. Furthermore, the proposed model could achieve a speed up in the order of 15–35 times through the parallelization of computationally intensive steps of the algorithm using GP-GPU and the CUDA-programming model in the MATLAB parallel computing environment. Finally, in terms of RS applications of land cover classification, the proposed method is reasonably better compared with some of the other methods in sparse domain. It gives the most accurate classified regions in terms of percent accuracy of average pixel counts of classified regions with those of the original image.

VII. Conclusion

This article presents an efficient SISR method based on JSR and adaptive dictionary learning for RS images. Reconstructed images are enriched with both local and nonlocal information. It reduces the high-computation costs of global dictionary training. The proposed algorithm with adaptive JSR technique is able to show competitive performances with several other sparse optimization and DL-based SR algorithms. Reasonable accelerations are also achieved through GPGPU-based parallel implementation. Finally, the high classification accuracy of the reconstructed images indicates the superiority of the proposed method. In future, as an extension of this work, robust feature extractions from MS images for learning the dictionaries may be explored, while a joint reconstruction problem can be formulated incorporating all the band images to enhance the spectral information as well.

References

[1] V. Starovoitov, A. Makarau, I. Zakharov, and D. Dovnar, "Multispectral image enhancement based on fusion and super-resolution," in *Proc. 15th Eur. Signal Process. Conf.*, 2007, pp. 2174–2178.

[2] X. X. Zhu and R. Bamler, "A sparse image fusion algorithm with application to pan-sharpening," *IEEE Trans. Geosci. Remote Sens.*, vol. 51, no. 5, pp. 2827–2836, May 2013.

[3] L. Liebel and M. Körner, "Single-image super resolution for multispectral remote sensing data using convolutional neural networks," *ISPRS - Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci.*, vol. XLI-B3, pp. 883–890, 2016.

[4] Y. Romano, J. Isidoro, and P. Milanfar, "RAISR: Rapid and accurate image super resolution," *IEEE Trans. Comput. Imag.*, vol. 3, no. 1, pp. 110–125, Mar. 2017.

[5] R. Fernandez-Beltran, P. Latorre-Carmona, and F. Pla, "Single-frame super-resolution in remote sensing: A practical overview," *Int. J. Remote Sens.*, vol. 38, no. 1, pp. 314–354, 2017.

[6] S. Farsiu, D. Robinson, M. Elad, and P. Milanfar, "Advances and challenges in super-resolution," *Int. J. Imag. Syst. Technol.*, vol. 14, no. 2, pp. 47–57, 2004.

[7] A. Marquina and S. J. Osher, "Image super-resolution by TV-regularization and bregman iteration," *J. Sci. Comput.*, vol. 37, no. 3, pp. 367–382, 2008.

[8] J. Sun, Z. Xu, and H.-Y. Shum, "Image super-resolution using gradient profile prior," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Anchorage, AK, USA, 2008, pp. 1–8.

[9] M. Elad, *Sparse and Redundant Representations: From Theory to Applications in Signal and Image Processing*, 1st ed. New York, NY, USA: Springer, 2010.

[10] I. Daubechies, M. Defrise, and C. De Mol, "An iterative thresholding algorithm for linear inverse problems with a sparsity constraint," *Commun. Pure Appl. Math.*, vol. 57, no. 11, pp. 1413–1457, 2004.

[11] D. L. Donoho and M. Elad, "On the stability of the basis pursuit in the presence of noise," *Signal Process.*, vol. 86, no. 3, pp. 511–532, 2006.

[12] R. Tibshirani, "Regression shrinkage and selection via the LASSO: A retrospective," *J. Roy. Statist. Soc. B*, vol. 73, no. 3, pp. 273–282, 2011.

[13] M. Aharon, M. Elad, and A. Bruckstein, "K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation," *IEEE Trans. Signal Process.*, vol. 54, no. 11, pp. 4311–4322, Nov. 2006.

[14] J. Yang, J. Wright, T. S. Huang, and Y. Ma, "Image super-resolution via sparse representation," *IEEE Trans. Image Process.*, vol. 19, no. 11, pp. 2861–2873, Nov. 2010.

[15] D. Glasner, S. Bagon, and M. Irani, "Super-resolution from a single image," in *Proc. IEEE 12th Int. Conf. Comput. Vis.*, Kyoto, Japan, 2009, pp. 349–356.

[16] W. Dong, L. Zhang, G. Shi, and X. Wu, "Image deblurring and super-resolution by adaptive sparse domain selection and adaptive regularization," *IEEE Trans. Image Process.*, vol. 20, no. 7, pp. 1838–1857, Jul. 2011.

[17] M. S. Moustafa, H. M. Ebied, A. K. Helmy, T. M. Nazamy, and M. F. Tolba, "Acceleration of super-resolution for multispectral images using self-example learning and sparse representation," *Comput. Elect. Eng.*, vol. 62, pp. 249–265, 2017.

[18] S. Gu, W. Zuo, Q. Xie, D. Meng, X. Feng, and L. Zhang, "Convolutional sparse coding for image super-resolution," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2015, pp. 1823–1831.

[19] P. Song, X. Deng, J. F. C. Mota, N. Deligiannis, P. L. Dragotti, and M. R. D. Rodrigues, "Multimodal image super-resolution via joint sparse representations induced by coupled dictionaries," *IEEE Trans. Comput. Imag.*, vol. 6, pp. 57–72, May 2019.

[20] K. Chang, P. L. K. Ding, and B. Li, "Single image super-resolution using collaborative representation and non-local self-similarity," *Signal Process.*, vol. 149, pp. 49–61, 2018.

[21] B. Hou, K. Zhou, and L. Jiao, "Adaptive super-resolution for remote sensing images based on sparse representation with global joint dictionary model," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 4, pp. 2312–2327, Apr. 2018.

[22] M. Cheng, C. Wang, and J. Li, "Single-image super-resolution in RGB space via group sparse representation," *IET Image Process.*, vol. 9, no. 6, pp. 461–467, 2015.

[23] J. Zhang, D. Zhao, and W. Gao, "Group-based sparse representation for image restoration," *IEEE Trans. Image Process.*, vol. 23, no. 8, pp. 3336–3351, Aug. 2014.

[24] W. Yang, J. Liu, S. Yang, and Z. Quo, "Image super-resolution via non-local similarity and group structured sparse representation," in *Proc. Vis. Commun. Image Process.*, 2015, pp. 1–4.

[25] S. Xu and F. Gao, "Single-image super resolution based on group sparse representation via GAUSSIAN," *Int. J. Circuits, Syst. Signal Process.*, vol. 11, pp. 118–128, 2017.

[26] S. Liu, G. Zhang, and W. Liu, "Group sparse representation based dictionary learning for SAR image despeckling," *IEEE Access*, vol. 7, pp. 30809–30817, 2019.

[27] Z. Zha, X. Yuan, B. Wen, J. Zhou, J. Zhang, and C. Zhu, "A benchmark for sparse coding: When group sparsity meets rank minimization," *IEEE Trans. Image Process.*, vol. 29, pp. 5094–5109, Mar. 2020.

[28] X. Lu, D. Wang, W. Shi, and D. Deng, "Group-based single image super-resolution with online dictionary learning," *EURASIP J. Adv. Signal Process.*, vol. 2016, no. 1, 2016, Art. no. 84.

[29] Z. Zha, X. Yuan, B. Wen, J. Zhang, J. Zhou, and C. Zhu, "Image restoration using joint patch-group-based sparse representation," *IEEE Trans. Image Process.*, vol. 29, pp. 7735–7750, Jul. 2020.

[30] E. Mikaeli, A. Aghagolzadeh, and M. Azghani, "Single-image super-resolution via patch-based and group-based local smoothness modeling," *Vis. Comput.*, vol. 36, no. 8, pp. 1573–1589, 2020.

[31] L. Gao, D. Hong, J. Yao, B. Zhang, P. Gamba, and J. Chanussot, "Spectral superresolution of multispectral imagery with joint sparse and low-rank learning," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 3, pp. 2269–2280, Mar. 2021.

[32] Z. Zha, B. Wen, X. Yuan, J. Zhou, and C. Zhu, "Image restoration via reconciliation of group sparsity and low-rank models," *IEEE Trans. Image Process.*, vol. 30, pp. 5223–5238, May 2021.

[33] J. Shi, X. Liu, Y. Zong, C. Qi, and G. Zhao, "Hallucinating face image by regularization models in high-resolution feature space," *IEEE Trans. Image Process.*, vol. 27, no. 6, pp. 2980–2995, Jun. 2018.

[34] H. Liu, R. Xiong, Q. Song, F. Wu, and W. Gao, "Image super-resolution based on adaptive joint distribution modeling," in *Proc. IEEE Vis. Commun. Image Process.*, 2017, pp. 1–4.
[35] S. Wang, T. Zhou, Y. Lu, and H. Di, " Contextual transformation network for lightweight remote-sensing image super-resolution," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, Dec. 2021, Art. no. 5615313.
[36] R. Dong, L. Mou, L. Zhang, H. Fu, and X. X. Zhu, "Real-world remote sensing image super-resolution via a practical degradation model and a kernel-aware network," *ISPRS J. Photogramm. Remote Sens.*, vol. 191, pp. 155–170, 2022.
[37] S. Lei and Z. Shi, "Hybrid-scale self-similarity exploitation for remote sensing image super-resolution," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, Apr. 2021, Art. no. 5401410.
[38] C. Dong, C. C. Loy, K. He, and X. Tang, "Learning a deep convolutional network for image super-resolution," in *Proc. Eur. Conf. Comput. Vis.*, 2014, pp. 184–199.
[39] J. Kim, J. K. Lee, and K. M. Lee, "Accurate image super-resolution using very deep convolutional networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 1646–1654.
[40] C. Tian, Y. Xu, W. Zuo, B. Zhang, L. Fei, and C.-W. Lin, "Coarse-to-fine CNN for image super-resolution," *IEEE Trans. Multimedia*, vol. 23, pp. 1489–1502, Jun. 2020.
[41] D. Zhang, J. Shao, X. Li, and H. T. Shen, "Remote sensing image super-resolution via mixed high-order attention network," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 6, pp. 5183–5196, Jun. 2021.
[42] T. Dai, J. Cai, Y. Zhang, S.-T. Xia, and L. Zhang, "Second-order attention network for single image super-resolution," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 11057–11066.
[43] W. Li et al., "Best-buddy GANs for highly detailed image super-resolution," in *Proc. AAAI Conf. Artif. Intell.*, 2022, 2021, pp. 1412–1420.
[44] H. Tan, H. Xiao, Y. Liu, M. Zhang, and B. Wang, "LASSO approximation and application to image super-resolution with CUDA acceleration," in *Proc. 2nd Int. Conf. Image. Vis. Comput.*, 2017, pp. 483–488.
[45] V. V. Attarde and A. Khaparde, "Super resolution of image using sparse representation of image patches with LASSO approximation on CUDA platform," in *Proc. 2nd Int. Conf. Intell. Comput. Control Syst.*, 2018, pp. 1533–1538.
[46] J. Xu, C. Qi, and Z. Chang, "Coupled K-SVD dictionary training for super-resolution," in *Proc. IEEE Int. Conf. Image Process.*, 2014, pp. 3910–3914.
[47] H. Lee, A. Battle, R. Raina, and A. Y. Ng, "Efficient sparse coding algorithms," in *Proc. 19th Int. Conf. Neural Inf. Process. Syst.*, 2006, pp. 801–808.
[48] B. Gao, P. Lan, X. Chen, L. Zhang, and F. Sun, "Accelerated split Bregman method for image compressive sensing recovery under sparse representation.," *KSII Trans. Internet Inf. Syst.*, vol. 10, no. 6, pp. 2748–2766, 2016.
[49] B. Deka and S. Datta, *Fast Algorithms for Compressed Sensing MRI Reconstruction*. Berlin, Germany: Springer, 2019, pp. 31–74, doi: 10.1007/978-981-13-3597-6_3.
[50] C. Dong, C. C. Loy, K. He, and X. Tang, "Image super-resolution using deep convolutional networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, no. 2, pp. 295–307, Feb. 2016.
[51] Z. Lin et al., "Revisiting RCAN: Improved training for image super-resolution," 2022, *arXiv:2201.11279*.

Bhabesh Deka (Senior Member, IEEE) received the B.E. degree in electronics and telecommunication engineering from Assam Engineering College, Guwahati, India, in 1999, the M.Tech. degree in electronics design and technology from Tezpur University, Tezpur, India, in 2001, and the Ph.D. degree in electronics and electrical engineering from the Indian Institute of Technology Guwahati, Guwahati, India, in 2011.

He is currently a Professor with the Department of Electronics and Communication Engineering (ECE), Tezpur University, where he leads Intelligent Imaging and Vision Research Laboratory, Department of ECE. His major research interests include image processing, particularly inverse ill-posed problems, computer vision, compressive sensing magnetic resonance imaging, and machine learning for biomedical signal/image analysis.

Dr. Deka is a Fellow of the Institution of Electronics and Telecommunication Engineers and a Life Member of the Indian Unit of Pattern Recognition and Artificial Intelligence affiliated with the International Association for Pattern Recognition.

Helal Uddin Mullah received the B.Eng. degree in electronics and telecommunication engineering from Assam Engineering College, Guwahati, India, in 2012, and the M.Tech. degree in electronics and communication engineering from North-Eastern Hill University, Shillong, India, in 2015. He is currently working toward the Ph.D. degree in electronics and communication engineering from Tezpur University, Tezpur, India.

His research interests include sparse representation for solving ill-posed inverse problems in imaging technology, computer vision, AI-driven intelligent systems, etc. He is a member of International Association for the Engineers and a regular member of Machine Intelligence Research Laboratory, USA.

Trishna Barman received the B.Tech. degree in electronics and communication engineering from Tezpur University, Tezpur, India, in 2013, and the M.Tech. degree in VLSI design from the North Eastern Regional Institute of Science and Technology, Nirjuli, India, in 2017. She is currently working toward the Ph.D. degree in electronics and communication engineering from Tezpur University, Tezpur, India.

Her research interests include remote sensing image processing, computer vision, parallel computing, and deep learning.

Sumit Datta (Member, IEEE) received the B.Tech. degree in electronics and communication engineering from the National Institute of Technology Agartala, Agartala, India, in 2011, and the M.Tech. and Ph.D. degrees in electronics and communication engineering from Tezpur University, Tezpur, India, in 2014 and 2019, respectively.

He is an Assistant Professor with the School of Electronic Systems and Automation, Digital University Kerala (DUK), Thiruvananthapuram, India. Prior to joining DUK, he was a Postdoctoral Fellow with the Department of Electronics and Electrical Engineering, Indian Institute of Technology Guwahati, India. His research interests include biomedical signal/image processing, compressed sensing MRI, super-resolution, and medical image analysis using deep learning.