Multi-Modal Convolutional Dictionary Learning

Fangyuan Gao, Xin Deng, *Member, IEEE*, Mai Xu, *Senior Member, IEEE*, Jingyi Xu, and Pier Luigi Dragotti, *Fellow, IEEE*

***Abstract*— Convolutional dictionary learning has become increasingly popular in signal and image processing for its ability to overcome the limitations of traditional patch-based dictionary learning. Although most studies on convolutional dictionary learning mainly focus on the unimodal case, real-world image processing tasks usually involve images from multiple modalities, e.g., visible and near-infrared (NIR) images. Thus, it is necessary to explore convolutional dictionary learning across different modalities. In this paper, we propose a novel multi-modal convolutional dictionary learning algorithm, which efficiently correlates different image modalities and fully considers neighborhood information at the image level. In this model, each modality is represented by two convolutional dictionaries, in which one dictionary is for common feature representation and the other is for unique feature representation. The model is constrained by the requirement that the convolutional sparse representations (CSRs) for the common features should be the same across different modalities, considering that these images are captured from the same scene. We propose a new training method based on the alternating direction method of multipliers (ADMM) to alternatively learn the common and unique dictionaries in the discrete Fourier transform (DFT) domain. We show that our model converges in less than 20 iterations between the convolutional dictionary updating and the CSRs calculation. The effectiveness of the proposed dictionary learning algorithm is demonstrated on various multimodal image processing tasks, achieves better performance than both dictionary learning methods and deep learning based methods with limited training data.**

***Index Terms*— Multi-modal dictionary learning, convolutional sparse coding, image denoising.**

I. Introduction

RECENTLY, multimodal image processing has attracted increasing interest from the signal and image processing communities. In many practical scenarios, one scene can be represented by many images from different modalities. For example, in virtual reality applications, RGB-D camera is typically used to simultaneously capture the color image and depth image [1]. In remote sensing, many hyperspectral images with different wavelengths correspond to the same Earth observation [2]. For object detection in a low-light scenario, it is normal to have a pair of visible and near-infrared (NIR) images. These images from multiple modalities represent the same scene but often offer complementary information; thus, they are often complementary to each other. However, due to the limitations of the acquisition devices, some of these multimodal images suffer severe distortions, e.g., noise contamination, low resolution, blurred edges, and missing regions. Therefore, an efficient multimodal representation model is needed to guide the restoration of the damaged modality.

Sparse representation (SR) and dictionary learning have long been used to represent signals and images [3]–[8]. With SR and dictionary learning, one signal is expected to be represented as a linear combination of a small number of atoms in an overcomplete dictionary. To represent multimodal data, Kwon *et al.* [9] proposed learning an independent dictionary for each modality and assumed that all dictionaries share the same sparse representations to link different modalities. Assuming that $\boldsymbol{x}$ and $\boldsymbol{y}$ denote the image patches from two different modalities and that their corresponding dictionaries are $\boldsymbol{D}_x$ and $\boldsymbol{D}_y$, [9] modeled the relationship between the two modalities as follows,

$$\begin{bmatrix} \boldsymbol{x} \\ \boldsymbol{y} \end{bmatrix} = \begin{bmatrix} \boldsymbol{D}_x \\ \boldsymbol{D}_y \end{bmatrix} \boldsymbol{\theta}. \tag{1}$$

Here, $\boldsymbol{\theta}$ is the sparse representation, which is the same for the two modalities. Since $\boldsymbol{x}$ and $\boldsymbol{y}$ are stacked together, $\boldsymbol{D}_x$ and $\boldsymbol{D}_y$ can be regarded as a large dictionary which can be learned as a whole. However, the above model represents each modality by one dictionary, which actually only considers the common parts between different modalities, but ignores the unique parts. To overcome this drawback, Zhang *et al.* [10] proposed representing each modality by two dictionaries, as follows:

$$\begin{bmatrix} \boldsymbol{x} \\ \boldsymbol{y} \end{bmatrix} = \begin{bmatrix} \boldsymbol{D}_c & \boldsymbol{D}_u & \boldsymbol{0} \\ \boldsymbol{D}_c & \boldsymbol{0} & \boldsymbol{D}_u \end{bmatrix} \begin{bmatrix} \boldsymbol{\theta}_c \\ \boldsymbol{\theta}_{u_x} \\ \boldsymbol{\theta}_{u_y} \end{bmatrix}, \tag{2}$$

in which $\boldsymbol{x}$ and $\boldsymbol{y}$ are represented by a common dictionary $\boldsymbol{D}_c$ and a unique dictionary $\boldsymbol{D}_u$. For the common dictionary $\boldsymbol{D}_c$, the sparse representation $\boldsymbol{\theta}_c$ is the same for both $\boldsymbol{x}$ and $\boldsymbol{y}$. For the unique dictionary $\boldsymbol{D}_u$, the sparse representations $\boldsymbol{\theta}_{u_x}$ and $\boldsymbol{\theta}_{u_y}$ are different for $\boldsymbol{x}$ and $\boldsymbol{y}$. Compared to [9], the model in [10] is more reasonable since the unique features between two modalities are represented more accurately. However,

Manuscript received May 26, 2021; revised October 25, 2021 and December 10, 2021; accepted December 28, 2021. Date of publication January 13, 2022; date of current version January 21, 2022. This work was supported in part by NSFC under Grant 62050175, Grant 62001016, and Grant 61922009; and in part by the Beijing Natural Science Foundation under Grant JQ20020. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Nikos Deligiannis. *(Corresponding author: Xin Deng.)*

Fangyuan Gao and Xin Deng are with the School of Cyber Science and Technology, Beihang University, Beijing 100191, China (e-mail: cindydeng@buaa.edu.cn).

Mai Xu and Jingyi Xu are with the Department of Electrical Information Engineering, Beihang University, Beijing 100191, China.

Pier Luigi Dragotti is with the Department of Electrical and Electronic Engineering, Imperial College London, London SW7 2AZ, U.K.

Digital Object Identifier 10.1109/TIP.2022.3141251

Zhang *et al.* [10] only decoupled the unique sparse representations, i.e., $\boldsymbol{\theta}_{u_x}$ and $\boldsymbol{\theta}_{u_y}$ are different, while the dictionaries are still the same for $\boldsymbol{x}$ and $\boldsymbol{y}$. Different from [10], Song *et al.* [11] proposed using different common and unique dictionaries for the two modalities, as follows:

$$\begin{bmatrix} \boldsymbol{x} \\ \boldsymbol{y} \end{bmatrix} = \begin{bmatrix} \boldsymbol{D}_{c_x} & \boldsymbol{D}_{u_x} & \boldsymbol{0} \\ \boldsymbol{D}_{c_y} & \boldsymbol{0} & \boldsymbol{D}_{u_y} \end{bmatrix} \begin{bmatrix} \boldsymbol{\theta}_c \\ \boldsymbol{\theta}_{u_x} \\ \boldsymbol{\theta}_{u_y} \end{bmatrix}. \tag{3}$$

In Eq. (3), to represent $\boldsymbol{x}$ and $\boldsymbol{y}$, not only the sparse representations but also the dictionaries are different. This makes the model of [11] more accurate than that of [9] and [10].

However, the learned dictionaries of the aforementioned models are redundant, i.e., the dictionary atom has many shifted copies. In addition, the whole image needs to be divided into overlapped patches to process each patch independently. The final result is obtained by assembling all patches together and averaging the overlapped pixels among adjacent patches. The overlap-averaging mechanism has several disadvantages. First, it ignores an important constraint in solving the patch estimation problem, i.e., pixels in the overlapped area of adjacent patches should be exactly the same. Second, the patch dividing and aggregation make the image processing with low efficiency. As an alternative representation, convolutional sparse coding (CSC) overcomes the above drawbacks, which decomposes the whole image into several sparse feature maps by convolutional filters. The CSC model is spatially invariant, e.g., a learned filter of a specific edge orientation can represent all edges of the same orientation in the image. In addition, it avoids dividing the image into overlapped patches and can naturally utilize the consistency prior.

Recently, Marivani *et al.* [12] proposed using convolutional sparse representation to model different image modalities at the image level. They assume that different modalities have their own convolutional dictionary which shares the same convolutional sparse representations. Inspired by Song *et al.* [11], Deng *et al.* [13] proposed to represent each image modality by two convolutional dictionaries, in which one dictionary is for common feature representation and the other is for unique feature representation. However, neither [12] nor [13] aims to solve the dictionary learning problem for multi-modal images. Instead, they both turn the multimodal dictionary model into a neural network by a deep unfolding strategy. Thus, they are essentially deep learning-based methods, which usually require a large amount of training data to achieve good performance. Without sufficient training data, the performance of these methods can be significantly degraded, as demonstrated in the experimental results in Section V. To bridge the gap, we propose in this paper a novel approach to solve the multi-modal convolutional dictionary learning problem through traditional optimization, and successfully apply it in various multi-modal image processing tasks.

The main contributions of this paper are as follows.

- We propose a new dictionary training algorithm to solve the multimodal convolutional dictionary learning problem set in [13]. Different from [13], which solves the problem by neural networks, we solve it through traditional optimization, and it is demonstrated to perform better than [13] with limited training data. The training process is performed in the discrete Fourier transform (DFT) domain and is composed of two stages: convolutional dictionary updating and convolutional sparse coefficient (CSR) calculation.
- We develop several multi-modal image restoration and fusion algorithms based on the proposed multi-modal dictionary learning, which are demonstrated to achieve superior results than the traditional dictionary learning based methods both quantitatively and qualitatively.

The remainder of this paper is organized as follows. In Section II, we review related work on both traditional and multimodal dictionary learning. The proposed method is introduced in Section III. The multi-modal image processing applications are introduced in Section IV. Finally, Section V discusses the experimental results, and Section VI concludes this paper.

II. Related Works

In this section, we first review works on traditional sparse representation methods, and then move to the convolutional sparse representation. Finally, we review the works on multimodal image representaion methods, including both traditional and convolutional multimodal sparse models.

A. Traditional Sparse Representation

Considering a signal $\boldsymbol{x} \in \mathbb{R}^N$ and an overcomplete dictionary $\boldsymbol{D} \in \mathbb{R}^{N \times M} (N \ll M)$, the sparse representation (SR) model aims to approximate $\boldsymbol{x}$ using a small number of atoms in dictionary $\boldsymbol{D}$, i.e., $\boldsymbol{x} \simeq \boldsymbol{D}\boldsymbol{\alpha}$. Here, the vector $\boldsymbol{\alpha}$ contains all the coefficients to represent the signal $\boldsymbol{x}$. Since dictionary $\boldsymbol{D}$ is redundant, vector $\boldsymbol{\alpha}$ is not always unique. The SR model usually assumes that $\boldsymbol{\alpha}$ should have the fewest nonzero elements, which can be mathematically achieved by solving the following optimization problem:

$$\min_{\boldsymbol{\alpha}} \|\boldsymbol{\alpha}\|_0, \quad \|\boldsymbol{x} - \boldsymbol{D}\boldsymbol{\alpha}\|_2 \leq \epsilon. \tag{4}$$

Here, $\|\cdot\|_0$ is the ℓ_0 norm, which indicates the number of nonzero elements in vector $\boldsymbol{\alpha}$. The above optimization is NP-hard for unstructured dictionaries. For this reason, classical solutions are based on convex relaxation, e.g., basis pursuit [14], or on greedy methods such as orthogonal matching pursuit (OMP) [15]. In the SR model, the dictionary $\boldsymbol{D}$ plays an important role. Many methods have been proposed to learn the dictionary to represent a certain set of training signals. The representative algorithms include K-SVD [16], MOD [17], and online dictionary learning [18]. The SR model has been widely used to solve inverse problems, such as image denoising, image super-resolution, and image inpainting, etc.

B. Convolutional Sparse Representation

Instead of sparsely representing a vector by the linear combination of dictionary atoms as in Eq. (4), an alternative SR model, namely convolutional sparse representation (CSR), models an entire image as a sum over a set of convolutions

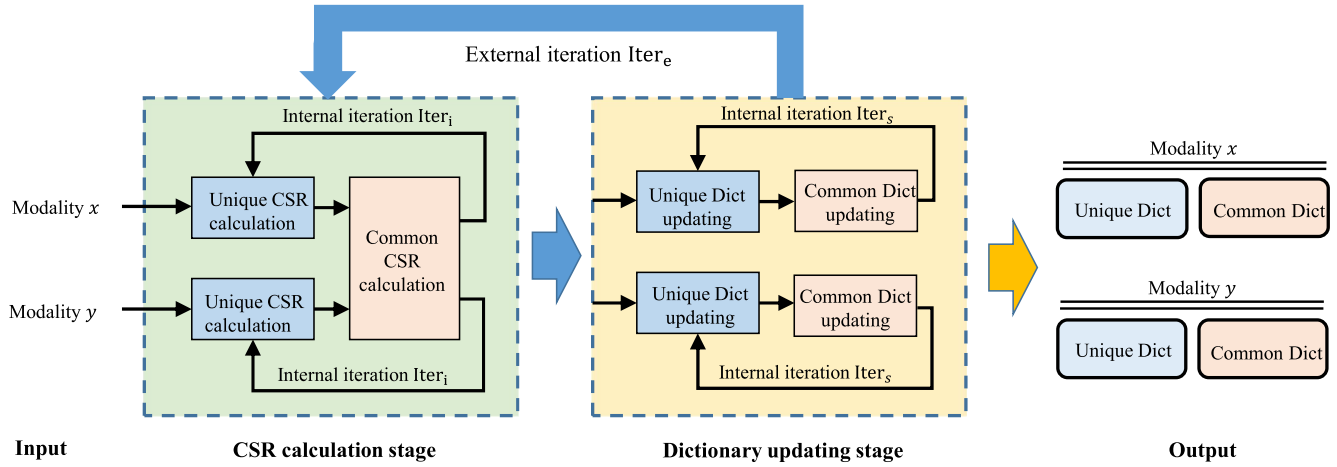

Fig. 1. Illustration of the training process of the proposed multimodal convolutional dictionary learning (MCDL) algorithm, which is mainly composed of the convolutional sparse representation (CSR) calculation and dictionary updating stages. The final output is a set of four convolutional dictionaries, with each modality having one unique and one common convolutional dictionary.

of coefficient maps with their corresponding dictionary filters [19]. The number of coefficient maps is the same as the number of dictionary filters, and the size of each coefficient map is the same as that of the image. Similar to the SR model, the CSR model also assumes that the coefficient map should be sparse, and it can be calculated via the following:

$$\underset{\{\boldsymbol{\alpha}_m\}}{\operatorname{argmin}} \frac{1}{2}\|\boldsymbol{x} - \sum_{m=1}^{M} \boldsymbol{d}_m * \boldsymbol{\alpha}_m\|_2 + \lambda \sum_{m} \|\boldsymbol{\alpha}_m\|_1. \tag{5}$$

Here, $\{\boldsymbol{d}_m\}$ is a set of M dictionary filters, and $\{\boldsymbol{\alpha}_m\}$ is a set of convolutional sparse representations. The size of $\boldsymbol{\alpha}_m$ is the same as the size of $\boldsymbol{x}$. To solve the optimization problem in (5), Bristow *et al.* [20] proposed a method based on the alternating direction method of multipliers (ADMM) [21], which operates in the DFT domain to speed up the calculation process. Later, Wohlberg *et al.* [19] further improved over [20] by applying Sherman-Morrison formula to solve the large liner system for computational advantage. Later, to reduce the computational complexity, Wang [22] proposed an online convolutional sparse coding algorithm. The CSR model has been successfully applied in many image processing tasks [23]–[29], such as the single image super-resolution task [23], image fusion [24], image decomposition [25], [28] and medical image processing [27], [29]. In [23], compared to the traditional SR model, the image super-resolution performance was improved by 0.26 dB in PSNR by using the CSR model for 2× upscaling. In [24], it is also demonstrated that CSR-based image fusion performs better than traditional SR-based image fusion.

C. Multimodal Image Representation

Multimodal dictionary learning has widely been used for many multimodal image processing tasks, such as multimodal image super-resolution [9], [11], image transformation [30], and image separation [31], etc. For image transformation, Wang *et al.* [30] proposed a semi-coupled dictionary learning (SCDL) algorithm, which correlates two modalities by assuming that the sparse representation of the source modality can be mapped to that of the target modality through a linear transform. For the multimodal retrieval task, Zhuang *et al.* [32] proposed a supervised coupled dictionary learning algorithm called $SliM^2$, which assumes that the sparse representations of different modalities can be mapped to each other, i.e., bidirectional mappings exist between any two modalities. For person re-identification, Jing *et al.* [33] proposed a semi-coupled low-rank discriminant dictionary learning algorithm, which is similar to SCDL [30] but adds a new discriminant constraint on the sparse representations to ensure the re-identification accuracy. For image classification tasks, Bahrampour *et al.* [34] proposed a multimodal task-driven dictionary learning algorithm under the group sparsity prior to enforcing collaborations among multiple modalities. For image separation, Deligiannis *et al.* [31] proposed a coupled dictionary learning algorithm, which assumes that part of the mixed modality shares the same sparse representations with the two separated modalities, and part of the mixed modality has its own sparse representation.

However, all the methods mentioned above learn dictionaries at the patch level, which makes it difficult to identify the global dependencies across modalities at the image level. Compared to the patch-based dictionary model, the convolutional dictionary model is able to capture the global dependencies across different modalities well [23], [35]. Specifically, for the task of multimodal image super-resolution, Marivani *et al.* [12] proposed a multimodal convolutional sparse representation model in which each image modality has its own convolutional dictionary, and different modalities are required to have the same convolutional sparse codes. Different from [12], where each modality has only one dictionary, Deng *et al.* [13] proposed a multimodal dictionary model in which each modality has two convolutional dictionaries for more accurate decoupling and representation.

In addition to multimodal image representation, there is another closely related research area namely multichannel image representation. Specifically, Wohlberg [36] proposed

the first convolutional dictionary learning algorithm for multichannel signals. There are two models proposed in [36] to relate multichannel signals, i.e., single channel and multichannel dictionary model. In single channel dictioanry model, each channel is represented by the same single-channel dictionary but with different sparse coefficient. In contrast, in multichannel dictionary model, all channels share the same representation, but with different dictionaries. Later, Hu *et al.* [37] proposed to model the RGB+NIR image reconstruction as a multichannel convolutional sparse coding problem, and used ADMM algorihtm to solve this problem. For video super-resolution, Barajas-Solano *et al.* [38] proposed to model the spectral video sequence by multichannel convolutional sparse coding, and achieved state-of-the-art spectral video super-resolution performance. La Tour *et al.* [39] first attempted to model electromagnetic brain signals by multivariate convolutional sparse coding, which is able to learn both the prototypical temporal waveforms and the associated spatial patterns.

III. Multi-Modal Convolutional Dictionary Learning

A. Motivation and Problem Statement

In multi-modal image processing, images from different modalities are often captured from the same scene. Thus, they are expected to share some common features, e.g., edges and shapes. However, since they are usually captured using different sensors, they also contain unique features, which are different across modalities. Take the RGB and depth images for example, the discontinuities in the depth image are clearly related to the edges in the RGB image. However, RGB images contain texture information, which does not exist in depth images. To this end, following [13], we represent each modality with two set of filters, i.e., common and unique filters.

Suppose that $\boldsymbol{x}$ and $\boldsymbol{y}$ are two image modalities, we represent each of them using two sets of convolutional filters, as follows:

$$\begin{aligned} \boldsymbol{x} &= \sum_{k=1}^{K} \boldsymbol{d}_k * \boldsymbol{c}_k + \sum_{m=1}^{M} \boldsymbol{e}_m * \boldsymbol{u}_m, \\ \boldsymbol{y} &= \sum_{k=1}^{K} \boldsymbol{h}_k * \boldsymbol{c}_k + \sum_{n=1}^{N} \boldsymbol{g}_n * \boldsymbol{v}_n, \end{aligned} \tag{6}$$

where $\{\boldsymbol{d}_k\}_{k=1}^{K}$ and $\{\boldsymbol{h}_k\}_{k=1}^{K}$ are the common filters of $\boldsymbol{x}$ and $\boldsymbol{y}$, respectively, $\{\boldsymbol{e}_m\}_{m=1}^{M}$ and $\{\boldsymbol{g}_n\}_{n=1}^{N}$ are the unique filters of $\boldsymbol{x}$ and $\boldsymbol{y}$, respectively. Here, K, M and N are the number of different filters. In addition, $\boldsymbol{c}_k$ is the common convolutional sparse representation (CSR) shared by the common filters $\{\boldsymbol{d}_k\}_{k=1}^{K}$ and $\{\boldsymbol{h}_k\}_{k=1}^{K}$, $\boldsymbol{u}_m$ and $\boldsymbol{v}_n$ are the unique CSRs of the unique filters $\{\boldsymbol{e}_m\}_{m=1}^{M}$ and $\{\boldsymbol{g}_n\}_{n=1}^{N}$, respectively. Note that the $\boldsymbol{x}$ and $\boldsymbol{y}$ are usually not directly decomposed into CSRs. The normal way is filtering them by a lowpass filter and decomposing the high-pass component into CSRs [19], [24]. The low-pass and high-pass components are decomposed by solving the following optimization problem:

$$\underset{\boldsymbol{x}_l}{\operatorname{argmin}} \|\boldsymbol{x} - \boldsymbol{x}_l\|_F^2 + \gamma \left(\left\|\boldsymbol{f}_h * \boldsymbol{x}_l\right\|_F^2 + \left\|\boldsymbol{f}_v * \boldsymbol{x}_l\right\|_F^2\right), \tag{7}$$

where $\boldsymbol{x}_l$ is the low-pass component, and $(\boldsymbol{x} - \boldsymbol{x}_l)$ is the high-pass component. The $\boldsymbol{f}_h = [-1, 1]$ and $\boldsymbol{f}_v = [-1, 1]^T$ are the horizontal and vertical gradient operators, respectively, and γ is the regularization parameter which is set to 5. This is a Tikhonov regularization problem, which can be solved by the fast Fourier transform. In this paper, we used the SPORCO library [40] to solve this problem. The $\boldsymbol{x}$ and $\boldsymbol{y}$ are independently low-pass filtered to obtain the high-pass components. This can be regarded as a pre-processing step. In the following, we perform joint representation of the highpass components.

The optimization formulation of our multi-modal convolutional dictionary learning is established as follows:

$$\begin{aligned} \underset{\substack{\{\boldsymbol{d}_k, \boldsymbol{e}_m, \boldsymbol{h}_k, \boldsymbol{g}_n, \\ \boldsymbol{c}_k^j, \boldsymbol{u}_m^j, \boldsymbol{v}_n^j\}}}{\operatorname{argmin}} & \frac{1}{2} \sum_j \left\| \boldsymbol{x}_j - \left(\sum_k \boldsymbol{d}_k * \boldsymbol{c}_k^j + \sum_m \boldsymbol{e}_m * \boldsymbol{u}_m^j\right) \right\|_2^2 \\ & + \frac{1}{2} \sum_j \left\| \boldsymbol{y}_j - \left(\sum_k \boldsymbol{h}_k * \boldsymbol{c}_k^j + \sum_n \boldsymbol{g}_n * \boldsymbol{v}_n^j\right) \right\|_2^2 \\ & + \gamma_c \sum_j \sum_k \left\|\boldsymbol{c}_k^j\right\|_1 + \gamma_u \sum_j \sum_m \left\|\boldsymbol{u}_m^j\right\|_1 \\ & + \gamma_v \sum_j \sum_n \left\|\boldsymbol{v}_n^j\right\|_1, \\ s.t., \quad & \|\boldsymbol{d}_k\|_2^2 = 1, \quad \|\boldsymbol{e}_m\|_2^2 = 1, \quad \|\boldsymbol{h}_k\|_2^2 = 1, \quad \left\|\boldsymbol{g}_n\right\|_2^2 = 1. \end{aligned} \tag{8}$$

Here, $\{\boldsymbol{x}_j\}_{j=1}^{J}$ and $\{\boldsymbol{y}_j\}_{j=1}^{J}$ are the training samples, $\boldsymbol{c}_k^j$ is the k-th common CSR of $\boldsymbol{x}_j$ and $\boldsymbol{y}_j$, $\boldsymbol{u}_m^j$ is the m-th unique CSR of $\boldsymbol{x}_j$, and $\boldsymbol{v}_n^j$ is the n-th unique CSR of $\boldsymbol{y}_j$. The first two items guarantee that the CSRs can well reconstruct the source images, and the remained items are used to constrain the sparsity of the CSRs.

Next, we focus on learning these dictionary filters and CSRs. We solve Eq. (8) in two steps as shown in Fig. 1. In the first step, we fix all the dictionary filters to update the CSRs, and in the second step, we fix all the CSRs to update the dictionary filters.

B. Updating the CSRs

By fixing all the convolutional dictionary filters, we have the following problem to solve:

$$\begin{aligned} \underset{\{\boldsymbol{c}_k^j, \boldsymbol{u}_m^j, \boldsymbol{v}_n^j\}}{\operatorname{argmin}} & \frac{1}{2} \sum_j \left\| \boldsymbol{x}_j - \left(\sum_k \boldsymbol{d}_k * \boldsymbol{c}_k^j + \sum_m \boldsymbol{e}_m * \boldsymbol{u}_m^j\right) \right\|_2^2 \\ & + \frac{1}{2} \sum_j \left\| \boldsymbol{y}_j - \left(\sum_k \boldsymbol{h}_k * \boldsymbol{c}_k^j + \sum_n \boldsymbol{g}_n * \boldsymbol{v}_n^j\right) \right\|_2^2 \\ & + \gamma_c \sum_j \sum_k \left\|\boldsymbol{c}_k^j\right\|_1 + \gamma_u \sum_j \sum_m \left\|\boldsymbol{u}_m^j\right\|_1 \\ & + \gamma_v \sum_j \sum_n \left\|\boldsymbol{v}_n^j\right\|_1, \end{aligned} \tag{9}$$

Here, we have three variables to optimize, and our solution is to update each variable in an alternating way. First, the common CSR $\boldsymbol{c}_k^j$ and unique CSR $\boldsymbol{u}_m^j$ are fixed to update the other unique CSR $\boldsymbol{v}_n^j$. Then, $\boldsymbol{c}_k^j$ and $\boldsymbol{v}_n^j$ are fixed to update $\boldsymbol{u}_m^j$. Finally, the unique CSRs $\boldsymbol{u}_m^j$ and $\boldsymbol{v}_n^j$ are fixed to update the common CSR $\boldsymbol{c}_k^j$.

Step 1: With $\boldsymbol{c}_k^j$ and $\boldsymbol{u}_m^j$ fixed, we can update $\boldsymbol{v}_n^j$ via the following:

$$\underset{\{\boldsymbol{v}_n^j\}}{\operatorname{argmin}} \frac{1}{2}\sum_j \left\| \boldsymbol{y}_j - (\sum_k \boldsymbol{h}_k * \boldsymbol{c}_k^j + \sum_n \boldsymbol{g}_n * \boldsymbol{v}_n^j) \right\|_2^2 + \gamma_v \sum_j \sum_n \left\| \boldsymbol{v}_n^j \right\|_1 \tag{10}$$

Since $\boldsymbol{c}_k^j$ is fixed, we can denote $\boldsymbol{y}'_j = \boldsymbol{y}_j - \sum_k \boldsymbol{h}_k * \boldsymbol{c}_k^j$, and then Eq. (10) can be simplified as follows:

$$\underset{\{\boldsymbol{v}_n^j\}}{\operatorname{argmin}} \frac{1}{2}\sum_j \left\| \boldsymbol{y}'_j - \sum_n (\boldsymbol{g}_n * \boldsymbol{v}_n^j) \right\|_2^2 + \gamma_v \sum_j \sum_n \left\| \boldsymbol{v}_n^j \right\|_1 \tag{11}$$

This is a standard convolutional sparse coding problem, which can be solved by ADMM in DFT domain. We use the software SPORCO [41] to solve this problem.

Step 2: With $\boldsymbol{c}_k^j$ and $\boldsymbol{v}_n^j$ fixed, we can update $\boldsymbol{u}_m^j$ via solving the following optimization problem:

$$\underset{\{\boldsymbol{u}_m^j\}}{\operatorname{argmin}} \frac{1}{2}\sum_j \left\| \boldsymbol{x}_j - (\sum_k \boldsymbol{d}_k * \boldsymbol{c}_k^j + \sum_m \boldsymbol{e}_m * \boldsymbol{u}_m^j) \right\|_2^2 + \gamma_u \sum_j \sum_m (\left\| \boldsymbol{u}_m^j \right\|_1) \tag{12}$$

By denoting $\boldsymbol{x}'_j = \boldsymbol{x}_j - \sum_k \boldsymbol{d}_k * \boldsymbol{c}_k^j$, we can turn Eq. (12) into

$$\underset{\{\boldsymbol{u}_m^j\}}{\operatorname{argmin}} \frac{1}{2}\sum_j \left\| \boldsymbol{x}'_j - \sum_m (\boldsymbol{e}_m * \boldsymbol{u}_m^j) \right\|_2^2 + \gamma_u \sum_j \sum_m \left\| \boldsymbol{u}_m^j \right\|_1 \tag{13}$$

Similar to Eq. (10), Eq. (13) is also a standard convolutional sparse coding problem.

Step 3: With $\boldsymbol{u}_m^j$ and $\boldsymbol{v}_n^j$ fixed, the updating of $\boldsymbol{c}_k^j$ is more challenging. Specifically, we have the following coupled optimization problem:

$$\underset{\{\boldsymbol{c}_k^j\}}{\operatorname{argmin}} \frac{1}{2}\sum_j \left\| \boldsymbol{x}_j - (\sum_k \boldsymbol{d}_k * \boldsymbol{c}_k^j + \sum_m \boldsymbol{e}_m * \boldsymbol{u}_m^j) \right\|_2^2 + \frac{1}{2}\sum_j \left\| \boldsymbol{y}_j - (\sum_k \boldsymbol{h}_k * \boldsymbol{c}_k^j + \sum_n \boldsymbol{g}_n * \boldsymbol{v}_n^j) \right\|_2^2 + \gamma_c \sum_j \sum_k \left\| \boldsymbol{c}_k^j \right\|_1, \tag{14}$$

Since $\boldsymbol{u}_m^j$ and $\boldsymbol{v}_n^j$ are fixed, we can denote $\boldsymbol{x}''_j = \boldsymbol{x}_j - \sum_m \boldsymbol{e}_m * \boldsymbol{u}_m^j$ and $\boldsymbol{y}''_j = \boldsymbol{y}_j - \sum_n \boldsymbol{g}_n * \boldsymbol{v}_n^j$, and Eq. (14) can be simplified as follows:

$$\underset{\{\boldsymbol{c}_k^j\}}{\operatorname{argmin}} \frac{1}{2}\sum_j \left\| \boldsymbol{x}''_j - \sum_k \boldsymbol{d}_k * \boldsymbol{c}_k^j \right\|_2^2 + \frac{1}{2}\sum_j \left\| \boldsymbol{y}''_j - \sum_k \boldsymbol{h}_k * \boldsymbol{c}_k^j \right\|_2^2 + \gamma_c \sum_j \sum_k \left\| \boldsymbol{c}_k^j \right\|_1. \tag{15}$$

By defining linear operators $\boldsymbol{D}_k$ and $\boldsymbol{H}_k$ such that $\boldsymbol{D}_k \boldsymbol{c}_k^j = \boldsymbol{d}_k * \boldsymbol{c}_k^j$ and $\boldsymbol{H}_k \boldsymbol{c}_k^j = \boldsymbol{h}_k * \boldsymbol{c}_k^j$, we can change Eq. (15) as follows:

$$\underset{\{\boldsymbol{c}_k^j\}}{\operatorname{argmin}} \frac{1}{2}\sum_j \left\| \boldsymbol{x}''_j - \sum_k \boldsymbol{D}_k \boldsymbol{c}_k^j \right\|_2^2 + \frac{1}{2}\sum_j \left\| \boldsymbol{y}''_j - \sum_k \boldsymbol{H}_k \boldsymbol{c}_k^j \right\|_2^2 + \gamma_c \sum_j \sum_k \left\| \boldsymbol{c}_k^j \right\|_1. \tag{16}$$

Here, $\boldsymbol{D}_k$ is a Toeplitz matrix constructed by $\boldsymbol{d}_k$, as follows:

$$\boldsymbol{D}_k = \begin{bmatrix} \boldsymbol{d}_k(1) & 0 & 0 & \cdots \\ \vdots & \boldsymbol{d}_k(1) & 0 & \cdots \\ \boldsymbol{d}_k(n) & \vdots & \boldsymbol{d}_k(1) & \cdots \\ 0 & \boldsymbol{d}_k(n) & \vdots & \cdots \\ 0 & 0 & \boldsymbol{d}_k(n) & \cdots \\ \vdots & \vdots & \vdots & \vdots \end{bmatrix} \tag{17}$$

Here, $\boldsymbol{d}_k(1)$ is the i-th element in $\boldsymbol{d}_k$. We then define

$$\boldsymbol{c}^j = \begin{pmatrix} \boldsymbol{c}_0^j \\ \boldsymbol{c}_1^j \\ \vdots \end{pmatrix} \quad and \quad \boldsymbol{C} = (\boldsymbol{c}^0 \quad \boldsymbol{c}^1 \quad \cdots), \tag{18}$$

$$\boldsymbol{X} = (\boldsymbol{x}''_0\ \boldsymbol{x}''_1 \cdots) \quad and \quad \boldsymbol{Y} = (\boldsymbol{y}''_0 \quad \boldsymbol{y}''_1 \quad \cdots), \tag{19}$$

and

$$\boldsymbol{D} = (\boldsymbol{D}_0\ \boldsymbol{D}_1 \cdots) \quad and \quad \boldsymbol{H} = (\boldsymbol{H}_0 \quad \boldsymbol{H}_1 \quad \cdots). \tag{20}$$

Then, we can rewrite Eq. (16) as follows:

$$\underset{\boldsymbol{C}}{\operatorname{argmin}} \frac{1}{2}\left\|\boldsymbol{X} - \boldsymbol{D}\boldsymbol{C}\right\|_F^2 + \frac{1}{2}\left\|\boldsymbol{Y} - \boldsymbol{H}\boldsymbol{C}\right\|_F^2 + \gamma_c \left\|\boldsymbol{C}\right\|_1 \tag{21}$$

ADMM Algorithm: To solve Eq. (21), we introduce an auxiliary variable $\boldsymbol{S}$ and have the following optimization problem,[1]

$$\underset{\boldsymbol{C}}{\operatorname{argmin}} \frac{1}{2}\left\|\boldsymbol{X} - \boldsymbol{D}\boldsymbol{C}\right\|_F^2 + \frac{1}{2}\left\|\boldsymbol{Y} - \boldsymbol{H}\boldsymbol{C}\right\|_F^2 + \lambda \left\|\boldsymbol{S}\right\|_1, \quad s.t., \ \boldsymbol{S} = \boldsymbol{C} \tag{22}$$

With the dual variable $\boldsymbol{U}$, the alternating direction method of multipliers (ADMM) algorithm [21] can solve (22) by splitting

[1]As demonstrated in [20] and [36], compared to the spatial domain methods, such as FISTA, the ADMM algorithm performed in the DFT domain gives more efficient solutions to the CSR problem with substantially computational advantage. Thus, we also adopt the ADMM framework to solve Eq. (21).

it into the following three sub-problems:

$$\boldsymbol{C}^{(t+1)} = \underset{\boldsymbol{C}}{\operatorname{argmin}} \|\boldsymbol{X} - \boldsymbol{D}\boldsymbol{C}\|_F^2 + \|\boldsymbol{Y} - \boldsymbol{H}\boldsymbol{C}\|_F^2 + \frac{\rho}{2}\left\|\boldsymbol{C} - \boldsymbol{S}^{(t)} + \boldsymbol{U}^{(t)}\right\|_F^2, \tag{23}$$

$$\boldsymbol{S}^{(t+1)} = \underset{\boldsymbol{S}}{\operatorname{argmin}} \frac{\rho}{2}\left\|\boldsymbol{C}^{(t+1)} - \boldsymbol{S} + \boldsymbol{U}^{(t)}\right\|_F^2 + \gamma_c \|\boldsymbol{S}\|_1, \tag{24}$$

$$\boldsymbol{U}^{(t+1)} = \boldsymbol{U}^{(t)} + \boldsymbol{C}^{(t+1)} - \boldsymbol{S}^{(t+1)}. \tag{25}$$

In sub-problem (23), to make the notations simple, we denote $\boldsymbol{P} = \boldsymbol{S}^{(t)} - \boldsymbol{U}^{(t)}$, and we then have the following problem:

$$\boldsymbol{C}^{(t+1)} = \underset{\boldsymbol{C}}{\operatorname{argmin}} \|\boldsymbol{X} - \boldsymbol{D}\boldsymbol{C}\|_F^2 + \|\boldsymbol{Y} - \boldsymbol{H}\boldsymbol{C}\|_F^2 + \frac{\rho}{2}\|\boldsymbol{C} - \boldsymbol{P}\|_F^2 \tag{26}$$

By taking derivative of (26) with respect to $\boldsymbol{C}$, we can have the following linear system:

$$(\boldsymbol{D}^H\boldsymbol{D} + \boldsymbol{H}^H\boldsymbol{H} + \rho\boldsymbol{I})\boldsymbol{C} = \boldsymbol{D}^H\boldsymbol{X} + \boldsymbol{H}^H\boldsymbol{Y} + \rho\boldsymbol{P}. \tag{27}$$

To solve Eq. (27) efficiently, following [42], we transform it into the DFT domain to obtain:

$$(\hat{\boldsymbol{D}}^H\hat{\boldsymbol{D}} + \hat{\boldsymbol{H}}^H\hat{\boldsymbol{H}} + \rho\boldsymbol{I})\hat{\boldsymbol{C}} = \hat{\boldsymbol{D}}^H\hat{\boldsymbol{X}} + \hat{\boldsymbol{H}}^H\hat{\boldsymbol{Y}} + \rho\hat{\boldsymbol{P}}. \tag{28}$$

Here, the variable with hat indicates its DFT transform. This can be solved using iterated Sherman-Morrison algorithm [43].

The sub-problem (24) can be easily solved by soft-thresolding, which has the following solution:

$$\boldsymbol{S}^{t+1} = \mathbb{S}_{\frac{\gamma_c}{\rho}}(\boldsymbol{C}^{(t+1)} + \boldsymbol{U}^{(t)}), \tag{29}$$

where $\mathbb{S}_{\frac{\gamma_c}{\rho}}$ is the soft-thresolding operator, which is defined as follows,

$$\mathbb{S}_{\frac{\gamma_c}{\rho}}(\boldsymbol{z}) = sign(\boldsymbol{z}) \odot max(0, |\boldsymbol{z}| - \frac{\gamma_c}{\rho}) \tag{30}$$

In ADMM, the subproblems (23), (24) and (25) are alternatively solved until meeting the maximum iteration number $Iter_a$.

C. Updating Dictionary Filters

By fixing all the common and unique CSRs learned in the last section, we can update the dictionary filters through the following optimization:

$$\underset{\{\boldsymbol{d}_k, \boldsymbol{e}_m, \boldsymbol{h}_k, \boldsymbol{g}_n\}}{\operatorname{argmin}} \frac{1}{2}\sum_j \left\|\boldsymbol{x}_j - (\sum_k \boldsymbol{d}_k * \boldsymbol{c}_k^j + \sum_m \boldsymbol{e}_m * \boldsymbol{u}_m^j)\right\|_2^2 + \frac{1}{2}\sum_j \left\|\boldsymbol{y}_j - (\sum_k \boldsymbol{h}_k * \boldsymbol{c}_k^j + \sum_n \boldsymbol{g}_n * \boldsymbol{v}_n^j)\right\|_2^2$$
$$s.t., \quad \|\boldsymbol{d}_k\|_2^2 = 1, \quad \|\boldsymbol{e}_m\|_2^2 = 1, \quad \|\boldsymbol{h}_k\|_2^2 = 1, \quad \|\boldsymbol{g}_n\|_2^2 = 1. \tag{31}$$

Since $\boldsymbol{d}_k$ and $\boldsymbol{e}_m$ only exist in the first term, $\boldsymbol{h}_k$ and $\boldsymbol{g}_n$ only exist in the second term, the updating of $\boldsymbol{d}_k$ and $\boldsymbol{e}_m$ can be independent from that of $\boldsymbol{h}_k$ and $\boldsymbol{g}_n$. Take the updating of $\boldsymbol{d}_k$ and $\boldsymbol{e}_m$ for example, we have the following problem:

$$\underset{\{\boldsymbol{d}_k, \boldsymbol{e}_m\}}{\operatorname{argmin}} \frac{1}{2}\sum_j \left\|\boldsymbol{x}_j - (\sum_k \boldsymbol{d}_k * \boldsymbol{c}_k^j + \sum_m \boldsymbol{e}_m * \boldsymbol{u}_m^j)\right\|_2^2$$
$$s.t., \quad \|\boldsymbol{d}_k\|_2^2 = 1, \quad \|\boldsymbol{e}_m\|_2^2 = 1. \tag{32}$$

To solve this problem, there are two possible solutions. One solution is to alternatively update the unique filters $\boldsymbol{e}_m$ and the common filters $\boldsymbol{d}_k$. The other solution is to append the dictionary filters $\boldsymbol{d}_k$ and $\boldsymbol{e}_m$ as a single dictionary, and use a standard CSC solver to obtain the dictionary filters. In this paper, we adopted the first solution to solve this problem, and regard the second solution as an interesting future work. Specifically, we first fix the unique filters $\boldsymbol{e}_m$ to update the common filters $\boldsymbol{d}_k$, and then fix the common filters $\boldsymbol{d}_k$ to update the unique filters $\boldsymbol{e}_m$.

Step 1: When $\boldsymbol{e}_m$ is fixed, we can re-write (32) as:

$$\underset{\{\boldsymbol{d}_k\}}{\operatorname{argmin}} \frac{1}{2}\sum_j \left\|(\boldsymbol{x}_j - \sum_m \boldsymbol{e}_m * \boldsymbol{u}_m^j) - \sum_k \boldsymbol{d}_k * \boldsymbol{c}_k^j\right\|_2^2$$
$$s.t., \quad \|\boldsymbol{d}_k\|_2^2 = 1. \tag{33}$$

When regarding $(\boldsymbol{x}_j - \sum_m \boldsymbol{e}_m * \boldsymbol{u}_m^j)$ as a whole, this is a typical convolutional dictionary learning problem, and we can use the convolutional constrained method of optimal directions (CCMOD) [19] to solve it.

Step 2: When $\boldsymbol{d}_k$ is fixed, we can update $\boldsymbol{e}_m$ through solving the following problem:

$$\underset{\{\boldsymbol{e}_m\}}{\operatorname{argmin}} \frac{1}{2}\sum_j \left\|(\boldsymbol{x}_j - \sum_k \boldsymbol{d}_k * \boldsymbol{c}_k^j) - \sum_m \boldsymbol{e}_m * \boldsymbol{u}_m^j\right\|_2^2$$
$$s.t., \quad \|\boldsymbol{e}_m\|_2^2 = 1. \tag{34}$$

This is also a typical convolutional dictionary learning problem, which can be solved with the CCMOD algorithm [19]. The updating of $\boldsymbol{d}_k$ and $\boldsymbol{e}_m$ is performed in turn with a pre-defined maximum iteration number $Iter_i$. In this paper, we set the $Iter_i = 10$. The same method can be used to update the dictionary filters $\boldsymbol{h}_k$ and $\boldsymbol{g}_n$, i.e., we first fix $\boldsymbol{h}_k$ to update $\boldsymbol{g}_n$, and then fix $\boldsymbol{g}_n$ to update $\boldsymbol{h}_k$.

IV. Applications

A. Cross-Field Image Denoising

For the task of cross-field image denoising, our aim is to eliminate the noise in one modality with the guidance of another modality. Suppose that $\boldsymbol{x}$ is the noisy modality and that $\boldsymbol{y}$ is the guidance modality, the data model can be formulated as follows:

$$\boldsymbol{x} = \sum_k \boldsymbol{d}_k * \boldsymbol{c}_k + \sum_m \boldsymbol{e}_m * \boldsymbol{u}_m + \epsilon,$$
$$\boldsymbol{y} = \sum_k \boldsymbol{h}_k * \boldsymbol{c}_k + \sum_n \boldsymbol{g}_n * \boldsymbol{v}_n, \tag{35}$$

where ϵ is the noise we need to remove from image $\boldsymbol{x}$. Given the known dictionary filters, we can model the cross-field

Algorithm 1 The Proposed MCDL Training Algorithm

Input: Multi-modal training samples $\{\boldsymbol{x}_j\}_{j=1}^{J}$ and $\{\boldsymbol{y}_j\}_{j=1}^{J}$.
Output: Convolutional dictionary filters $\{\boldsymbol{d}_k\}_{k=1}^{K}$, $\{\boldsymbol{h}_k\}_{k=1}^{K}$, $\{\boldsymbol{e}_m\}_{m=1}^{M}$, and $\{\boldsymbol{g}_n\}_{n=1}^{N}$.

for p = 1 to $Iter_e$ **do**
  Update convolutional sparse representations (CSRs)
  for q = 1 to $Iter_i$ **do**
    Step 1: Update the unique CSRs of one modality through Eq. (11).
    Step 2: Update the unique CSRs of another modality through Eq. (13).
    Step 3: Update the common CSRs using ADMM in DFT domain through Eqs. (22)-(29).
  end for
  Update dictionary filters
  for s = 1 to $Iter_s$ **do**
    Step 1: Update the common dictionary filters through Eq. (33).
    Step 2: Update the unique dictionary filters through Eq. (34).
  end for
end for
return $\{\boldsymbol{d}_k\}_{k=1}^{K}$, $\{\boldsymbol{h}_k\}_{k=1}^{K}$, $\{\boldsymbol{e}_m\}_{m=1}^{M}$, $\{\boldsymbol{g}_n\}_{n=1}^{N}$

denoising problem as follows,

$$\begin{aligned}\underset{\{\boldsymbol{c}_k,\boldsymbol{u}_m,\boldsymbol{v}_n\}}{\operatorname{argmin}} \; & \frac{1}{2}\left\|\boldsymbol{x}-(\sum_k \boldsymbol{d}_k*\boldsymbol{c}_k+\sum_m \boldsymbol{e}_m*\boldsymbol{u}_m)\right\|_2^2 \\ & +\frac{1}{2}\left\|\boldsymbol{y}-(\sum_k \boldsymbol{h}_k*\boldsymbol{c}_k+\sum_n \boldsymbol{g}_n*\boldsymbol{v}_n)\right\|_2^2 \\ & +\lambda_c\sum_k\|\boldsymbol{c}_k\|_1+\lambda_u\sum_m\|\boldsymbol{u}_m\|_1+\lambda_v\sum_n\|\boldsymbol{v}_n\|_1 .\end{aligned} \tag{36}$$

where λ_c, λ_u and λ_v are used to constrain the sparsity level of $\boldsymbol{c}_k$, $\boldsymbol{u}_m$ and $\boldsymbol{v}_n$, respectively. As we described before, the common CSR $\boldsymbol{c}_k$ is expected to capture the common features between two modalities, and obviously noise is not a feature shared between the noisy image $\boldsymbol{x}$ and clean image $\boldsymbol{y}$. Thus, it is expected that most of the noise will appear in the unique CSR $\boldsymbol{u}_m$ of $\boldsymbol{x}$, and the $\boldsymbol{u}_m$ should be more sparse to remove more noise. This can be achieved by assigning a larger value to λ_u and smaller value to λ_c and λ_v. After solving Eq. (36) to have $\boldsymbol{c}_k$ and $\boldsymbol{u}_m$, we obtain the denoised image as follows,

$$\widetilde{\boldsymbol{x}}=\sum_k \boldsymbol{d}_k*\boldsymbol{c}_k+\sum_m \boldsymbol{e}_m*\boldsymbol{u}_m. \tag{37}$$

B. Multi-Modal Image Fusion

Multi-modal image fusion aims to fuse two images from different modalities, e.g., a pair of visible and near-infrared images. To apply the proposed multi-modal convolutional dictionary learning algorithm in image fusion, we assume that $\boldsymbol{x}$ and $\boldsymbol{y}$ are two source images to be fused and model their relationship as in Eq. (6). The fusion procedure is as follows. Given the common and unique dictionaries, we first calculate the common and unique CSRs by solving the same optimization problem as the cross-field image denoising task in Eq. (36). Different from the cross-field image denoising, the λ_c, λ_u and λ_v in this case have the same value. Then, after we obtain the common CSR $\boldsymbol{c}_k$, unique CSRs $\boldsymbol{u}_m$ and $\boldsymbol{v}_n$, the fused image $\boldsymbol{f}$ can be calculated by the following formulation:

$$\begin{aligned}\boldsymbol{f} = & \underbrace{w_1\sum_k \boldsymbol{d}_k*\boldsymbol{c}_k+w_2\sum_k \boldsymbol{h}_k*\boldsymbol{c}_k}_{\text{common part shared by } \boldsymbol{x} \text{ and } \boldsymbol{y}}+\underbrace{\sum_m \boldsymbol{e}_m*\boldsymbol{u}_m}_{\text{unique part of } \boldsymbol{x}} \\ & +\underbrace{\sum_n \boldsymbol{g}_n*\boldsymbol{v}_n}_{\text{unique part of } \boldsymbol{y}} .\end{aligned} \tag{38}$$

Here, the fused image $\boldsymbol{f}$ is composed of three parts, including the common part shared by $\boldsymbol{x}$ and $\boldsymbol{y}$, the unique part of $\boldsymbol{x}$, and the unique part of $\boldsymbol{y}$. Here, the common part is calculated by a weighted sum of the common parts of $\boldsymbol{x}$ and $\boldsymbol{y}$, i.e., $w_1\sum_k \boldsymbol{d}_k*\boldsymbol{c}_k+w_2\sum_k \boldsymbol{h}_k*\boldsymbol{c}_k$. Note that the sum of w_1 and w_2 is required to be 1, i.e, $w_1+w_2=1$. As demonstrated in the experimental results, when both w_1 and w_2 are equal to 0.5, we can obtain the best fusion performance.

C. Cross-Field Image Deblurring

For this task, there actually exist three modalities, i.e., given the RGB image and blurred MS image, we aim to restore the original MS image. Their relationship is modeled as follows, where $\boldsymbol{x}$, $\boldsymbol{y}$ and $\boldsymbol{z}$ denote the blurred MS image, RGB image, and restored MS image, respectively.

$$\begin{aligned}\boldsymbol{x} &= \sum_{k=1}^{K}\boldsymbol{d}_k*\boldsymbol{c}_k+\sum_{m=1}^{M}\boldsymbol{e}_m*\boldsymbol{u}_m, \\ \boldsymbol{y} &= \sum_{k=1}^{K}\boldsymbol{h}_k*\boldsymbol{c}_k+\sum_{n=1}^{N}\boldsymbol{g}_n*\boldsymbol{v}_n, \\ \boldsymbol{z} &= \sum_{k=1}^{K}\boldsymbol{l}_k*\boldsymbol{c}_k+\sum_{m=1}^{M}\boldsymbol{p}_m*\boldsymbol{u}_m,\end{aligned} \tag{39}$$

In the training process, we first train the dictionaries and CSRs for $\boldsymbol{x}$ and $\boldsymbol{y}$ as described in the original manuscript. Then, with $\boldsymbol{c}_k$ and $\boldsymbol{u}_m$ obtained, we can calculate dictionary filters $\boldsymbol{l}_k$ and $\boldsymbol{p}_m$ using convolutional constrained method of optimal directions (CCMOD) method.

V. Experiments

In this section, we first introduce the implementation details to train the multi-modal dictionaries, and then perform several multi-modal image restoration and fusion tasks to show the effectiveness of the proposed MCDL algorithm. Please note that the experiments here are only intended to prove the concept of using such multi-modal dictionaries in multi-modal image processing tasks. Further work is required to fully deploy the proposed algorithm in large-scale image-processing applications.

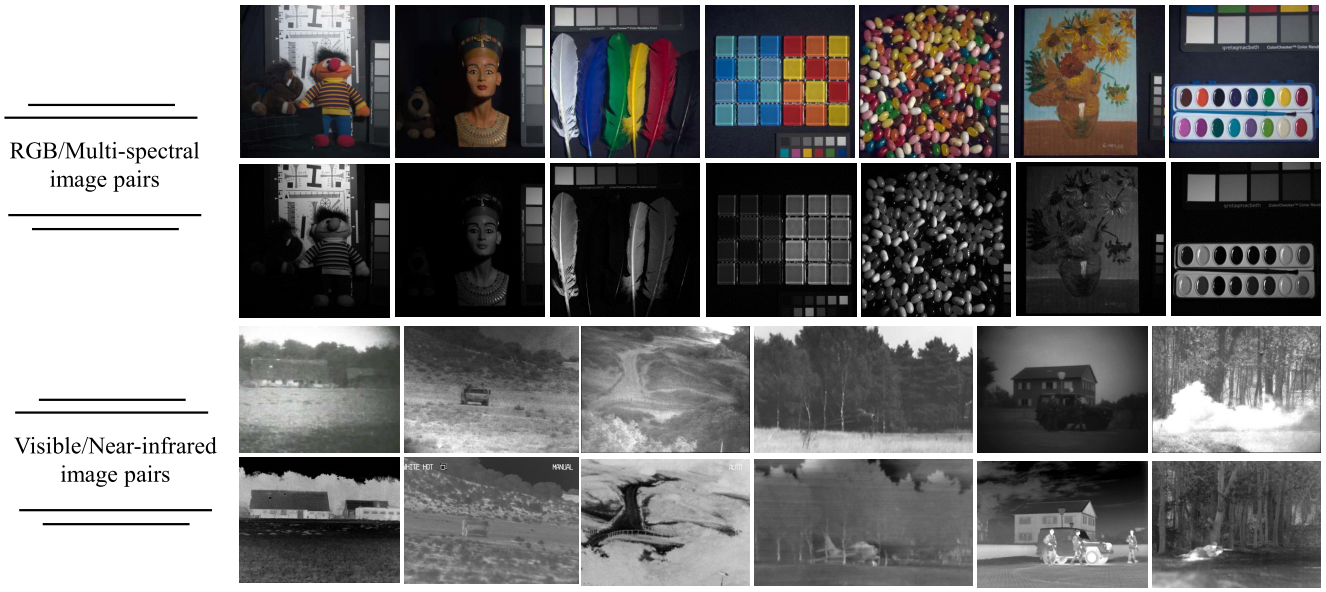

Fig. 2. Examples of the multi-modal testing images used in this paper. Here, the multi-spectral images are with 640 nm wavelength band.

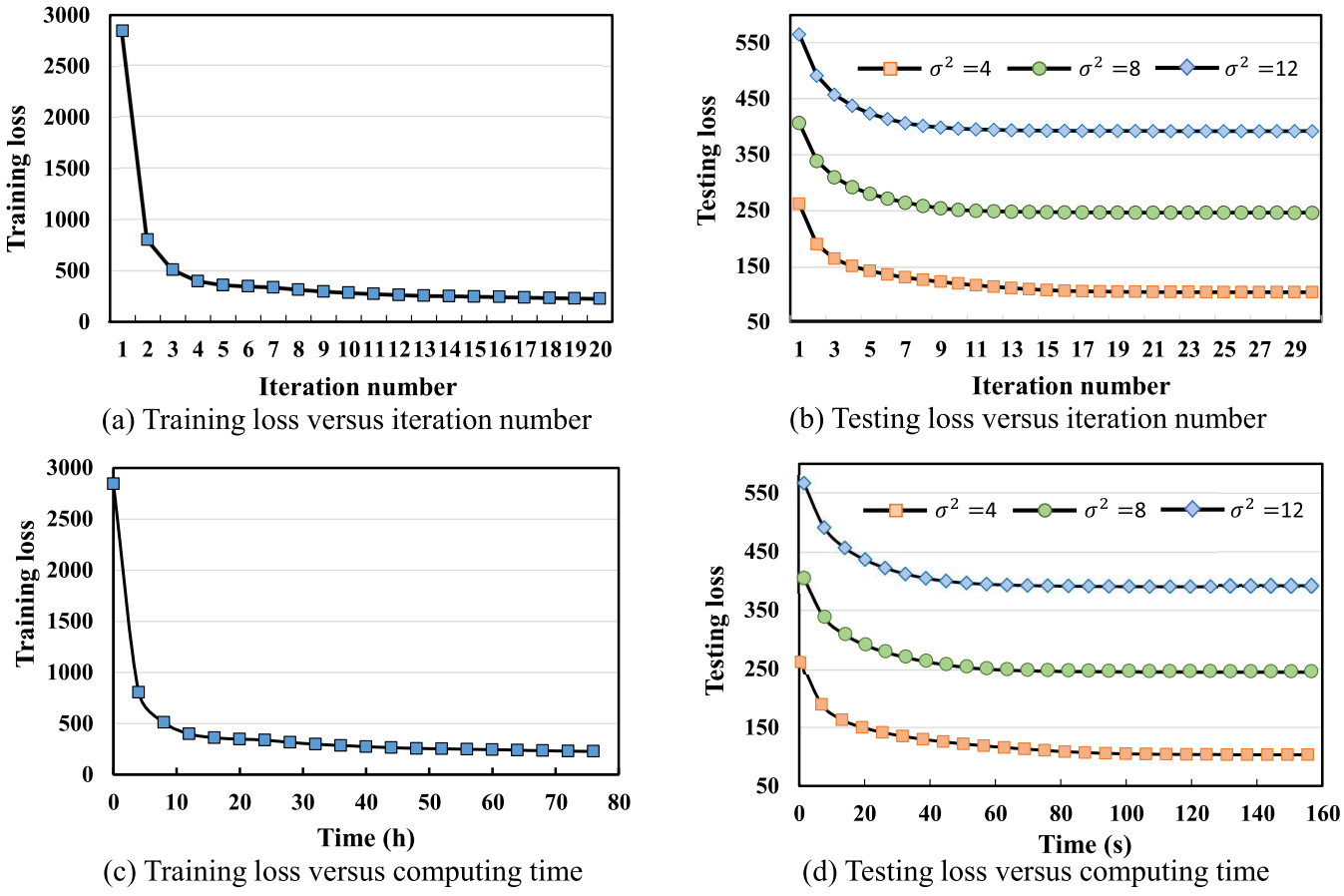

Fig. 3. (a) and (b) plot the curves of the training and testing losses with the increase of iteration number. (c) and (d) plot the real computing time in the training and testing processes in cross-filed image denoising. The testing image is of size 512 × 512.

A. Dictionary Learning Setup

1) Training Dataset: We train the multi-modal dictionaries using two different multi-modal datasets, including RGB/multi-spectral (MS) image pairs and visible/near-infrared (NIR) image pairs. The RGB/MS training images are from the Columbia multi-spectral database,[2] and RGB/NIR training images are from the EPFL RGB-NIR Scene database.[3] In this paper, we only use 10 pairs of multi-modal images for training.

[2]http://www.cs.columbia.edu/CAVE/databases/multispectral/
[3]http://ivrl.epfl.ch/supplementary_material/cvpr11/

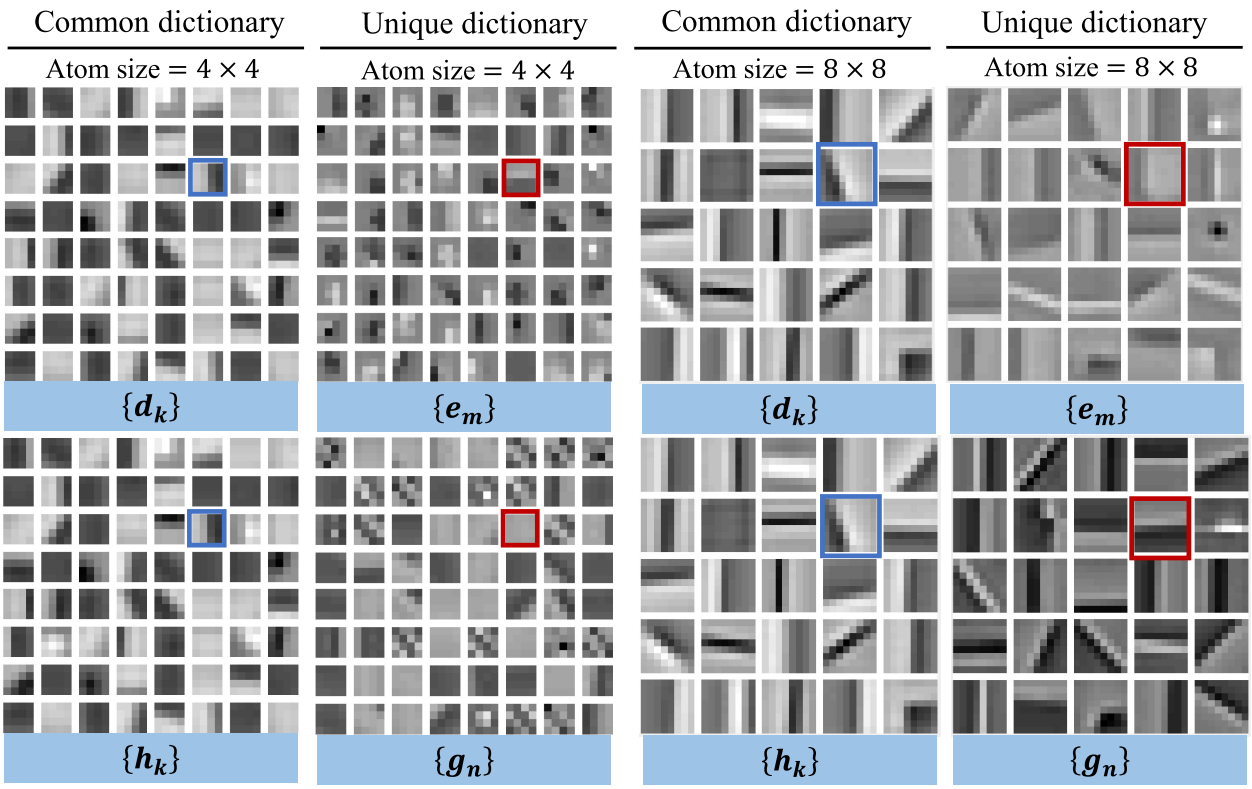

Fig. 4. Visualization of the common and unique convolutional dictionary atoms learned by the proposed MCDL algorithm for RGB/Multi-spectral scenario.

TABLE I

COMPUTING TIME OF DIFFERENT COMPONENTS FOR EACH INNER ITERATION IN TRAINING AND TESTING PROCESSES

Training process		
CSR Calculation	Unique CSR calculation	5.0 s
	Common CSR calculation	9.6 s
Dcitionary Updating	Common dictionary updating	36 m
	Unique dictionary updating	36 m
Testing process for a 512×512 image		
CSR Calculation	Unique CSR calculation	0.6 s
	Common CSR calculation	1.0 s

2) Testing Dataset: For cross-field image denoising and deblurring, we randomly select 7 pairs of RGB and multi-spectral images from Columbia multi-spectral database for testing. Note that the testing images are different from the training images. For image fusion, we randomly select 12 pairs of visible and NIR images from the TNO Image Fusion Dataset[4] for testing. Fig. 2 shows some examples of the testing images.

3) Training Parameters: In the training process, considering the computational complexity, the size of training patch is 64×64, and the total number of training patches is 3,000. Note that this does not violate the claim that the convolutional sparse coding is operated on the whole image. In the testing phase, the input to our model can be image of any size, which does not need to be split into patches. The number of dictionary filters in each dictionary is 32, and the filter size is 8×8. The γ_c, γ_u and γ_v are set to 0.1. The number of maximum internal iterations $Iter_i$ and $Iter_s$ is set to 10, and the number of maximum external iteration $Iter_e$ is set to 50 for iteratively updating dictionaries and sparse coefficients. In the ADMM algorithm, we set the maximum iteration number $Iter_a$ as 5. For the RGB images with 3 channels, we first turn them into YCbCr format and use only the Y channel for training.

[4]https://figshare.com/articles/dataset/TNO_Image_Fusion_Dataset/1008029

B. Analysis of Dictionary Learning Performance

1) Convergence Speed: The convergence speed is an important element to evaluate the efficiency of dictionary learning. Fig. 3 (a) and (c) show the convergence speed of our MCDL algorithm during the training process, in terms of iteration number and the real computing time, respectively. The computing time is recorded by implementing the experiments on a 64-bit Windows PC with Intel Core i7-4770 processor @3.40 GHz. It can be seen that the training loss decreases rapidly with the number of iterations, and nearly converges within 20 iterations. However, each iteration requires a large amount of training time, i.e., usually several hours, and the whole training process is very time-consuming. As presented in Table I, the dictionary updating process accounts for most computing time, while the CSR calculation only needs several seconds. Fig. 3 (b) and (d) show the loss curves when testing the cross-field image denoising task in terms of iteration number and computing time. We can see that the testing loss decreases rapidly and converges in less than 30 iterations. In addition, it has relatively fast computing time, i.e., each iteration only needs several seconds. In the testing process, we only need to calculate the CSRs, and the fast convergence speed verifies the effectiveness of our CSR calculation algorithm. For the other multi-modal datasets, we observe quite similar convergence performance.

TABLE II

CROSS-FIELD DENOISING PERFORMANCE IN TERMS OF PSNR (DB), WITH THE BEST RESULTS IN BOLD AND THE SECOND BEST RESULTS UNDERLINED

Noise level	Methods	Chart toy	Egyptian	Feathers	Glass tiles	Jelly beans	Oil painting	Paints	Average
σ^2=4	KSVD [4]	39.55	37.47	38.68	31.10	37.95	39.10	36.26	37.16
	BM3D [45]	39.47	41.43	39.34	38.95	38.18	38.13	38.97	39.21
	CDL [46]	39.42	45.71	41.88	39.04	38.10	**40.00**	42.67	40.97
	MuGIF [47]	39.83	43.50	40.04	39.07	38.04	35.07	40.48	39.43
	PGBL [48]	40.27	45.61	42.13	39.80	37.72	38.40	43.04	41.00
	Li *et al.* [49]	39.66	45.20	42.48	41.35	40.76	38.03	43.54	41.57
	Deng *et al.* [1]	40.22	44.14	41.81	39.96	39.28	39.18	41.93	40.93
	Ours	**41.95**	**46.51**	**43.07**	**41.79**	**41.23**	38.16	**43.60**	**42.33**
σ^2=8	K-SVD [4]	35.68	34.48	35.46	29.88	34.93	35.54	34.02	34.28
	BM3D [45]	37.95	40.90	38.30	37.70	35.73	36.03	38.19	37.83
	CDL [46]	37.73	43.54	39.67	37.22	36.41	**38.24**	39.91	38.96
	MuGIF [47]	**39.11**	42.73	39.27	38.42	36.98	34.86	39.47	38.69
	PGBL [48]	38.48	43.28	40.09	38.62	36.51	35.79	40.45	39.03
	Li *et al.* [49]	36.08	41.34	38.70	38.01	37.88	35.06	39.29	38.05
	Deng *et al.* [1]	37.25	42.22	38.77	36.86	36.72	36.20	38.55	38.08
	Ours	38.70	**43.92**	**40.18**	**38.75**	**38.27**	35.41	**40.90**	**39.44**
σ^2=12	K-SVD [4]	32.91	31.48	32.79	28.94	32.22	32.49	31.32	31.73
	BM3D [45]	36.66	40.09	37.25	36.62	34.28	34.82	37.38	36.73
	CDL [46]	36.44	41.87	38.15	35.94	34.98	**37.07**	37.96	37.49
	MuGIF [47]	**38.09**	41.76	38.16	**37.40**	35.59	34.50	38.20	37.67
	PGBL [48]	37.14	41.63	38.51	37.22	35.40	34.90	38.55	37.62
	Li *et al.* [49]	34.25	38.51	36.31	35.45	35.22	34.59	36.35	35.81
	Deng *et al.* [1]	36.23	41.26	38.03	35.94	35.76	34.59	37.52	37.05
	Ours	37.20	**42.50**	**38.66**	37.30	**36.27**	35.02	**38.78**	**37.96**

TABLE III

CROSS-FIELD DENOISING PERFORMANCE IN TERMS OF SSIM, WITH THE BEST RESULTS IN BOLD AND THE SECOND BEST RESULTS UNDERLINED

Noise level	Methods	Chart toy	Egyptian	Feathers	Glass tiles	Jelly beans	Oil painting	Paints	Average
σ^2=4	KSVD [4]	0.9647	0.8696	0.9739	0.9545	0.9891	0.9723	0.9799	0.9577
	BM3D [45]	0.9811	0.9710	0.9786	0.9803	0.9896	0.9756	0.9820	0.9797
	CDL [46]	0.9911	0.9886	0.9904	0.9888	0.9933	0.9836	0.9929	0.9898
	MuGIF [47]	0.9883	0.9825	0.9831	0.9875	0.9873	0.9373	0.9913	0.9800
	PGBL [48]	0.9935	0.9830	0.9923	0.9924	0.9930	**0.9764**	0.9936	0.9890
	Li *et al.* [49]	0.9835	0.9716	0.9853	0.9876	0.9896	0.9403	0.9924	0.9786
	Deng *et al.* [1]	0.9785	0.9677	0.9786	0.9773	0.9757	0.9555	0.9823	0.9737
	Ours	**0.9943**	**0.9892**	**0.9929**	**0.9931**	**0.9937**	0.9760	**0.9952**	**0.9907**
σ^2=8	K-SVD [4]	0.8810	0.6910	0.9085	0.9002	0.9612	0.9232	0.9463	0.8873
	BM3D [45]	0.9790	0.9755	0.9741	0.9765	0.9822	0.9550	0.9810	0.9748
	CDL [46]	0.9796	0.9760	0.9779	0.9784	0.9861	**0.9695**	0.9837	0.9787
	MuGIF [47]	0.9817	0.9732	0.9775	0.9817	0.9848	0.9335	0.9874	0.9743
	PGBL [48]	0.9870	0.9863	0.9848	0.9873	0.9863	0.9505	0.9916	0.9819
	Li *et al.* [49]	0.9595	0.9427	0.9622	0.9674	0.9746	0.8815	0.9754	0.9519
	Deng *et al.* [1]	0.9700	0.9696	0.9714	0.9720	0.9687	0.9581	0.9791	0.9698
	Ours	**0.9907**	**0.9872**	**0.9887**	**0.9894**	**0.9899**	0.9527	**0.9921**	**0.9844**
σ^2=12	K-SVD [4]	0.9708	0.5564	0.8252	0.8354	0.9272	0.8635	0.8870	0.8122
	BM3D [45]	0.9726	0.9694	0.9646	0.9700	0.9739	0.9381	0.9768	0.9665
	CDL [46]	0.9665	0.9585	0.9638	0.9682	0.9771	0.9556	0.9727	0.9661
	MuGIF [47]	0.9712	0.9631	0.9668	0.9709	0.9800	0.9241	0.9804	0.9652
	PGBL [48]	0.9799	0.9756	0.9772	0.9817	0.9791	0.9394	**0.9885**	0.9745
	Li *et al.* [49]	0.9106	0.8644	0.9143	0.9228	0.9395	0.8727	0.9297	0.9077
	Deng *et al.* [1]	0.9610	0.9568	0.9650	0.9656	0.9614	0.8713	0.9727	0.9506
	Ours	**0.9858**	**0.9834**	**0.9827**	**0.9840**	**0.9827**	**0.9444**	0.9867	**0.9785**

2) Visualization of Learned Dictionary Filters: Fig. 4 shows the common and unique filters learned using our MCDL algorithm for RGB/multi-spectral images with two different atom sizes, e.g., 4×4 and 8×8. As we can see, the common filters of these two modalities look similar to each other in the same location, while the unique filters are significantly different. This is consistent with our assumption that the common filters aim to capture the common feature among modalities, while the unique filters are to preserve the complementary features.

C. Application to Cross-Field Image Denoising

1) Additive Gaussian Noise: In this experiment, additive white Gaussian noise is added to the multi-spectral image with three different noise levels, i.e., $\sigma^2 = 4, 8, 12$. There is no noise in the RGB image, and it is used to guide the denoising of the multi-spectral image. The peak signal-to-noise ratio (PSNR) and SSIM [44] values are calculated between the noisy and denoised multi-spectral images to evaluated the denoising performance of our method. The larger value of PSNR and SSIM indicates better denoising performance.

2) Comparison Methods: We compare our method with the other two dictionary-based image denoising methods, including K-SVD denoising method [3] and coupled dictionary learning (CDL) based image denoising method [46], a bayesian learning based denoising method PGBL [48], BM3D [45], a mutual guided image filtering (MuGIF) method [47], and two deep learning

TABLE IV

RESULTS ON VISIBLE AND INFRARED IMAGE FUSION FOR DIFFERENT METHODS IN TERMS OF Q_{MI}, Q_{TE} AND SSIM

Q_{MI}	House	APC	Heather	Airplane	People	Tampax	Maninhuis	Meting	Soldier	Trench	Sandpath	Tank	Average
MST [51]	0.253	0.390	0.239	0.333	0.211	0.233	0.172	0.206	0.236	0.265	0.236	0.268	0.253
ADF [52]	0.294	**0.446**	0.229	0.393	0.247	0.275	**0.218**	0.245	0.284	0.291	0.240	0.177	0.278
CSMCA [53]	0.256	0.426	0.229	0.352	0.224	0.251	0.190	0.221	0.254	0.286	**0.241**	0.290	0.268
DIDFuse [54]	0.275	0.411	0.226	0.352	0.224	0.248	0.196	0.213	0.249	0.287	0.235	0.271	0.265
U2Fusion [55]	0.290	0.433	0.215	0.354	0.236	0.230	0.193	0.227	0.241	0.289	0.226	0.263	0.266
Ours	**0.315**	0.423	**0.249**	**0.404**	**0.258**	**0.288**	0.216	**0.246**	**0.285**	**0.312**	0.240	**0.296**	**0.294**
Q_{TE}	House	APC	Heather	Airplane	People	Tampax	Maninhuis	Meting	Soldier	Trench	Sandpath	Tank	Average
MST [51]	0.354	0.423	0.241	0.493	0.392	0.350	0.181	0.175	0.320	0.246	0.220	0.346	0.312
ADF [52]	0.353	**0.469**	**0.298**	0.513	**0.466**	**0.399**	**0.271**	**0.246**	0.341	0.291	**0.353**	0.372	**0.364**
CSMCA [53]	0.393	0.433	0.262	0.507	0.415	0.355	0.200	0.195	0.333	0.255	0.248	0.363	0.330
DIDFuse [54]	0.366	0.417	0.264	0.509	0.434	0.351	0.193	0.200	0.315	0.277	0.276	0.353	0.329
U2Fusion [55]	0.358	0.414	0.289	0.520	0.443	0.387	0.232	0.231	0.332	0.283	0.284	0.369	0.345
Ours	**0.425**	0.428	0.278	**0.535**	0.457	0.358	0.224	0.224	**0.347**	**0.291**	0.284	**0.398**	0.354
SSIM	House	APC	Heather	Airplane	People	Tampax	Maninhuis	Meting	Soldier	Trench	Sandpath	Tank	Average
MST [51]	1.9877	1.9974	1.9930	1.9964	1.9897	1.9931	1.9920	1.9928	1.9876	1.9941	1.9968	1.9854	1.9922
ADF [52]	1.9882	1.9976	1.9934	1.9967	1.9899	1.9936	1.9926	1.9932	1.9880	1.9944	1.9972	1.9809	1.9922
CSMCA [53]	1.9879	1.9975	1.9933	1.9965	1.9898	1.9933	1.9922	1.9929	1.9877	1.9943	1.9970	1.9858	1.9924
DIDFuse [54]	1.9865	1.9964	1.9929	1.9965	1.9889	1.9935	1.9918	1.9921	1.9864	1.9940	1.9965	1.9857	1.9918
U2Fusion [55]	1.9877	1.9969	1.9930	1.9965	1.9897	1.9934	1.9923	1.9928	1.9877	1.9940	1.9963	1.9804	1.9917
Ours	**1.9882**	**1.9976**	**1.9935**	**1.9967**	**1.9899**	**1.9937**	**1.9926**	**1.9932**	**1.9880**	**1.9944**	**1.9972**	**1.9866**	**1.9926**

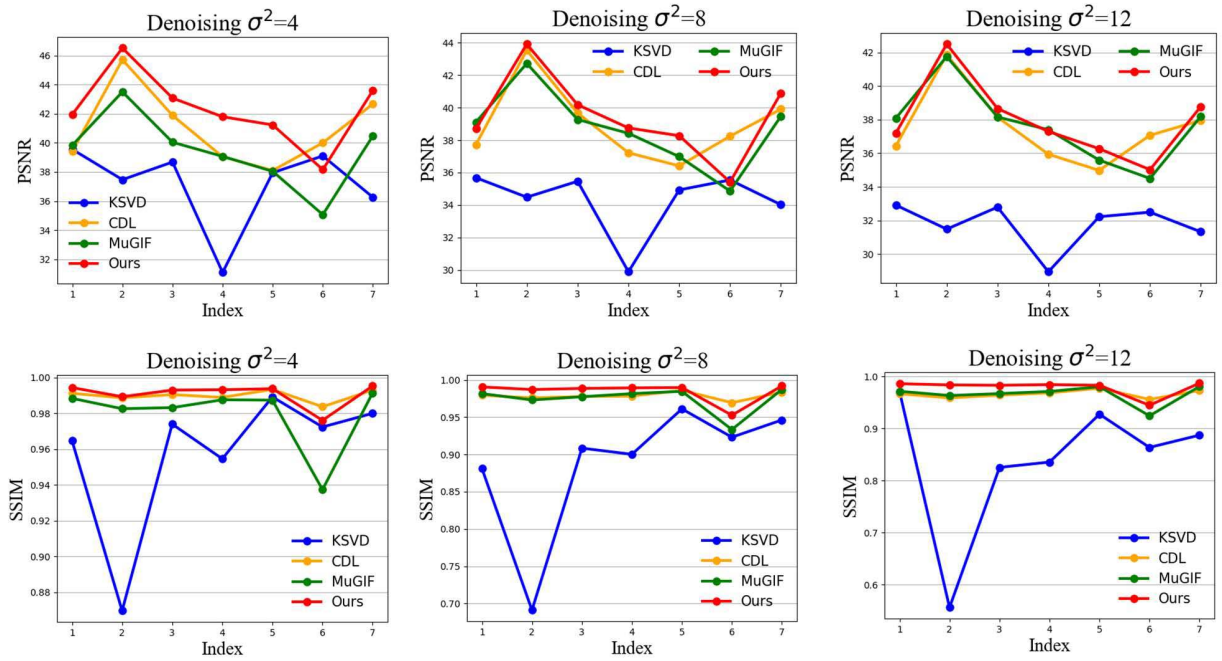

Fig. 5. The change of cross-field image denoising performance across different images for different noise levels, in terms of PSNR and SSIM.

based methods Deng *et al.* [13] and Li *et al.* [49]. For BM3D, the noise level is estimated by the method of Chen *et al.* [50]. For fair comparison, the deep learning based methods are re-trained using the same training dataset as ours, i.e., ten pairs of RGB/multispectral images.

3) Comparison Results: Tables II and III present the RGB guided multi-spectral image denoising results using our MCDL method for different noise levels, in terms of PSNR and SSIM, respectively. Here, for fair comparison, the number of dictionary atoms in both our and CDL methods are set to the same value as 32. As we can see from these two tables, our multi-modal dictionary learning algorithm achieves better denoising performance, compared to other two dictionary learning methods K-SVD and CDL. In addition, our method also outperforms the deep learning based methods Deng *et al.* [13] and Li *et al.* [49], when they are trained using the same small training dataset as ours. Normally, with sufficient training data, these deep learning based methods can perform significantly better than the dictionary learning based methods. However, in the case when the training data is limited, the dictionary learning based methods are preferred. This indicates the effectiveness of our MCDL algorithm. Fig. 5 visualizes the change of PSNR and SSIM across different images, and we can see that our line is normally above the other comparison methods, indicating that our approach achieves better cross-field denoising performance. Fig. 6 visualizes the denoised multi-spectral images using our method, and we can see that our method is able to eliminate most of the noise.

TABLE V

THE RGB GUIDED MS IMAGE DEBLURRING RESULTS OF OUR METHOD WITH DIFFERENT BLUR KERNELS IN PSNR (DB)

Blur kernel	Methods	Chart toy	Egyptian	Feathers	Glass tiles	Jelly beans	Oil painting	Paints	Average
3×3	Blurred	33.36	40.92	36.72	31.85	33.44	33.57	37.45	35.33
	Restored	38.85	46.03	40.20	35.71	38.09	40.88	40.93	**40.10**
5×5	Blurred	29.39	37.07	31.85	27.54	28.78	31.45	31.18	31.04
	Restored	34.42	41.60	35.44	31.24	33.59	36.45	36.31	**35.58**
7×7	Blurred	27.46	35.09	29.42	25.64	26.43	31.50	28.01	29.08
	Restored	32.23	39.85	33.46	29.48	31.44	34.57	34.29	**33.62**

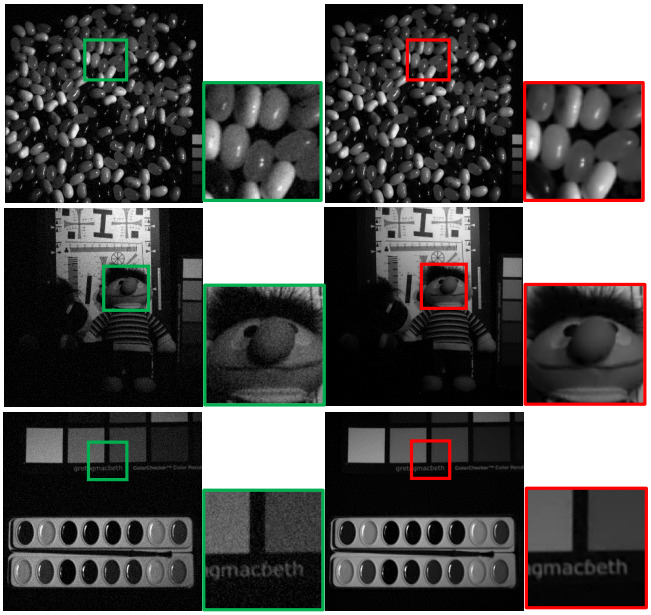

Fig. 6. Visualization of RGB guided multi-spectral image denoising results of our method. The first column is the noisy images and the second column shows the denoising results of our method. The first image is with noise level $\sigma^2 = 8$, while the second and third images are with $\sigma^2 = 12$.

TABLE VI

EFFECT OF SPARSITY CONSTRAINT ON CROSS-FIELD IMAGE DENOISING PERFORMANCE WITH $\sigma^2 = 12$

λ_c	λ_u	λ_v	PSNR (dB)	SSIM
0.10	0.10	0.10	37.54	0.9637
0.10	0.15	0.10	37.96	0.9785
0.10	0.20	0.10	37.84	0.9799
0.10	0.25	0.10	37.71	0.9798

TABLE VII

EFFECT OF ATOM SIZE ON CROSS-FIELD IMAGE DENOISING PERFORMANCE WITH $\sigma^2 = 12$

Atom size	PSNR (dB)	SSIM
4×4	37.03	0.9722
8×8	37.96	0.9785
12×12	37.39	0.9775

D. Application to Multi-Modal Image Fusion

We compare our method with other dictionary learning based image fusion methods, including MST [51], ADF [52], CSMCA [53], and two deep learning based methods DIDFuse [54] and U2Fusion [55]. For the two deep learning based methods, for fair comparison, we re-trained their networks using the same training data as ours. To evaluate the fusion performance, we adopt three widely used image fusion metrics, specifically Q_{MI} [56], Q_{TE} [57] and SSIM [44]. Here, Q_{MI} measures the mutual information between the fused image and the two source images, and Q_{TE} uses the Tsallis entropy to measure the degree of dependence between the fused and source images. The SSIM measures the structural similarity between the fused image and the source images. The larger value of the above three metrics indicates better fusion performance.

Table IV presents the fusion results of our method and the comparison methods in terms of Q_{MI}, Q_{TE} and SSIM. As we can see from this table, our method outperforms the other methods with larger value of Q_{MI} and SSIM. The Q_{TE} value of our method is slightly lower than ADF [52], but higher than other methods. The deep learning based methods DIDFuse [54] and U2Fusion [55] do not perform well with limited training data. The possible reason is that the optimization of the network parameters requires a large amount of training data, and the small training dataset makes the network fail to learn the optimal fusion strategy. This demonstrates the advantage of our MCDL algorithm over the deep learning based methods in the limited training data case.

Fig. 7 visualizes the fused images by different methods. As we can see from this figure, our method is able to generate the fused image with sharper edges, while the comparison methods have obvious halo effects around boundaries. In addition, Fig. 8 vividly shows the common and unique parts reconstructed by our method for a pair of visible and NIR images. It can be seen from Fig. 8 that their common parts are quite similar while their unique parts contain different features. For example, in the unique part of visible image, there is a tree which does not exist in the NIR image. In contrast, in the unique part of NIR image, there is a soldier standing besides the car which does not exist in the visible image. This indicates that our method is able to well reconstruct the common and unique parts from the images from different modalities, which helps improve the fusion performance.

E. Application to Cross-Field Image Deblurring

In this experiment, the Gaussian blur is added to the multi-spectral image with three different kernel sizes, i.e., 3×3, 5×5 and 7×7. The larger kernel size indicates the more severe blurring artifacts. There is no blur in the RGB image, and it is used to guide the deblurring of the multi-spectral image. The PSNR values are calculated between the blurred and restored multi-spectral images to evaluated the deblurring performance of our method.

Table V presents the RGB guided MS image deblurring results using our method for different Gaussian blurring kernel sizes. As we can see, our method is able to restore the MS image with different blurring kernel sizes. Fig. 9 shows the deblurring results by our method for 5×5 blurring kernel.

Fig. 7. Visual comparisons of visible and near-infrared image fusion results of different methods. (a) and (b) are the source images, (c) is MST [51] method, (d) is CSMCA [53] method, and (e) is our method.

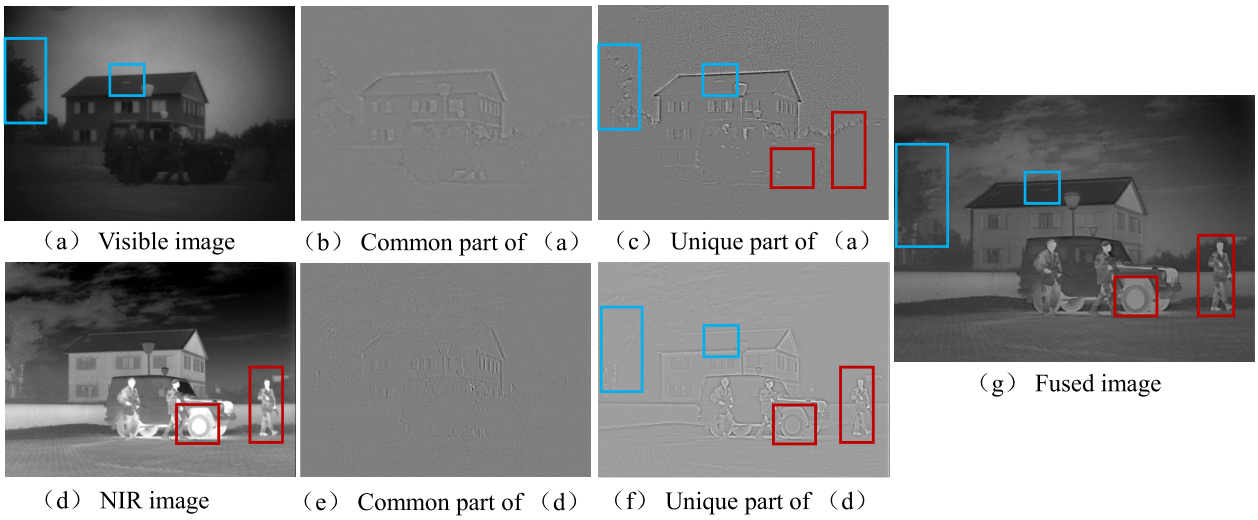

Fig. 8. Visualization of the common and unique parts reconstructed by our method in visible/NIR image fusion task.

TABLE VIII

THE FUSION PERFORMANCE WITH DIFFERENT WEIGHTS OF w_1 AND w_2 (OUR METHOD REQUIRES $w_1 + w_2 = 1$), WITH THE BEST RESULTS IN BOLD

weights	Our Method					Weighted average of images
	$w_1 = 0$ $w_2 = 1$	$w_1 = 0.3$ $w_2 = 0.7$	$w_1 = 0.5$ $w_2 = 0.5$	$w_1 = 0.7$ $w_2 = 0.3$	$w_1 = 0.5$ $w_2 = 0$	$w_1 = 0.5$ $w_2 = 0.5$
Q_{MI}	0.261	0.273	**0.294**	0.270	0.258	0.285
Q_{TE}	0.321	0.329	**0.354**	0.332	0.322	0.342
$SSIM$	1.9892	1.9920	**1.9926**	1.9919	1.9916	1.9923

As we can see from this figure, the restored image has clear edges, which demonstrates the effectiveness of our method.

F. Ablation Study

1) Effects of Sparsity Constraint: As we mentioned in Section IV, since most of the noise is in the unique part, a higher sparsity level should be given to the unique sparse representation u_m. In Eq. (36), the sparsity level of u_m is determined by λ_u, and the larger value of λ_u indicates the higher sparsity level. Table VI shows the effect of λ_u on the cross-field image denoising performance, with λ_c and λ_v fixed as 0.10. The testing dataset is the same as that in Section V-C. As we can see from this table, when λ_c, λ_u and λ_v are all set to 0.10, the PSNR value is 37.54 dB. With λ_u increased to 0.15, the PSNR value is increased to 37.96 dB. This is because more noise is removed with larger λ_u. However, when λ_u is further increased to 0.20 and 0.25, the PSNR value is

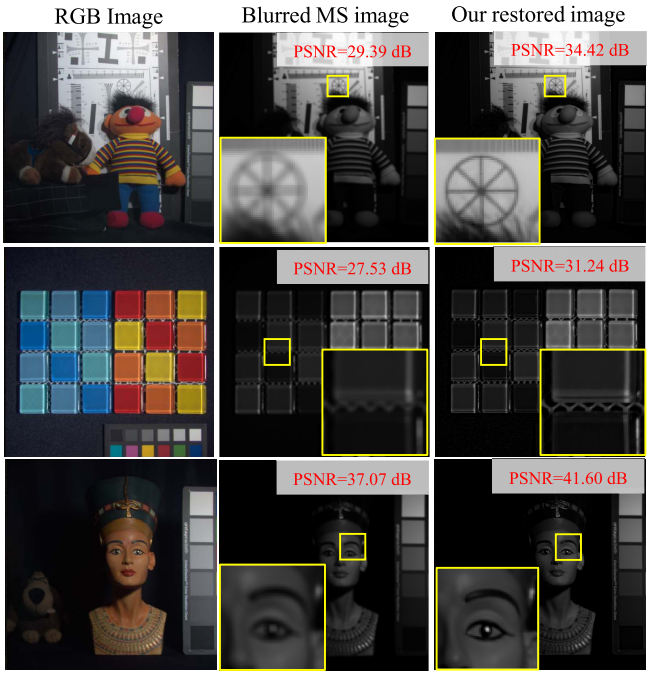

Fig. 9. Visualization of RGB guided multi-spectral image deblurring results of our method. The first column is the RGB images, the second column shows the blurred multi-spectral images, and the third column shows the restored images by our method.

decreased. The possible reason is that some image details are lost due to the higher sparsity constraint. This further indicates the effectiveness of our method in separating the common and unique features.

2) Effects of Atom Size: Table VII shows the denoising results with the dictionary atom as 4×4, 8×8 and 12×12, respectively. Here, the testing dataset is the same as that in Section V-C. As we can see from this table, the atom with size 8×8 leads to the best denoising performance with higher PSNR and SSIM values. Either smaller or larger atom size degrades the performance. The possible reason is that the small atom is not able to represent the global structural feature, while the large atom may lose some local details.

3) Effects of Weights in Image Fusion: Table VIII compares the fusion performance with different settings of w_1 and w_2 in Eq. (38). As can be seen from this table, when the weights of w_1 and w_2 are both set to 0.5, we can obtain the best fusion performance. We also compare with the weighted average of the two source images, which achieves its best performance when the two images are equally weighted ($w_1 = w_2 = 0.5$). As we can see from Table VIII, our method performs better than the best result of the weighted average of two source images. The possible reason is that the unique parts of the two source images are averaged, which decreases the fusion performance.

G. Missing Modality Discussion

In the testing phase, to make it consistent with the training phase, we normally need two modalities to achieve the multi-modal image restoration task. In the case when only one modality is available in the testing time, the task is actually turned from multi-modal image restoration to uni-modal image restoration. Take the RGB guided MS image deblurring as example, without the guidance of RGB image in the testing phase, our task becomes restoring the original MS image from a single blurred MS image. In other words, we only have the equations for $\boldsymbol{x}$ and $\boldsymbol{z}$ in Eq. (39), and they share both the same common and unique CSRs. Without $\boldsymbol{y}$, there is actually no need to distinguish between the common and unique CSRs between $\boldsymbol{x}$ and $\boldsymbol{z}$. Thus, we can combine the common and unique filters together to form a large dictionary for both $\boldsymbol{x}$ and $\boldsymbol{z}$. In the testing phase, to restore $\boldsymbol{z}$ from $\boldsymbol{x}$, we can calculate the CSRs with the large dictionary for $\boldsymbol{x}$, and then use it to restore $\boldsymbol{z}$.

VI. Conclusion and Future Work

In this paper, we introduced a multi-modal convolutional dictionary learning algorithm and, to the best of our knowledge, this is the first coupled convolutional dictionary learning algorithm towards multi-modal images. To verify the effectiveness of the proposed method, we apply it to various multi-modal image restoration and fusion tasks, including RGB guided multi-spectral image denoising/debluring, visible and NIR image fusion tasks. The numerical results demonstrate the effectiveness of the proposed method. As the first attempt on multi-modal convolutional dictionary learning, our work performs well but still has room for improvement, which will be our future work. First, the proposed common and unique CSR updating algorithm has nested iterations. One possible research direction is to find a more efficient algorithm to update the CSRs through ADMM Consensus framework, so that the nested iterations can be avoided. Second, in this paper, we only apply the MCDL algorithm in low level image processing tasks, like image denoising and fusion. One interesting future work is explore its application in high level tasks, e.g., image classification and image retrieval.

References

[1] F. Endres, J. Hess, J. Sturm, D. Cremers, and W. Burgard, "3-D mapping with an RGB-D camera," *IEEE Trans. Robot.*, vol. 30, no. 1, pp. 177–187, Feb. 2014.

[2] M. A. Veganzones, M. Simões, 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.

[3] M. Elad and M. Aharon, "Image denoising via sparse and redundant representations over learned dictionaries," *IEEE Trans. Image Process.*, vol. 15, no. 12, pp. 3736–3745, Dec. 2006.

[4] J. Xu, L. Zhang, and D. Zhang, "A trilateral weighted sparse coding scheme for real-world image denoising," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 20–36.

[5] 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.

[6] S. Yang, M. Wang, Y. Chen, and Y. Sun, "Single-image super-resolution reconstruction via learned geometric dictionaries and clustered sparse coding," *IEEE Trans. Image Process.*, vol. 21, no. 9, pp. 4016–4028, Sep. 2012.

[7] C. Zhang, J. Liu, Q. Tian, C. Xu, H. Lu, and S. Ma, "Image classification by non-negative sparse coding, low-rank and sparse decomposition," in *Proc. CVPR*, Jun. 2011, pp. 1673–1680.

[8] T. Zhang, B. Ghanem, S. Liu, C. Xu, and N. Ahuja, "Low-rank sparse coding for image classification," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2013, pp. 281–288.

[9] H. Kwon, Y.-W. Tai, and S. Lin, "Data-driven depth map refinement via multi-scale sparse representation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 159–167.

[10] Q. Zhang, Y. Fu, H. Li, and J. Zou, "Dictionary learning method for joint sparse representation-based image fusion," *Opt. Eng.*, vol. 52, no. 5, 2013, Art. no. 057006.

[11] 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, 2020.

[12] I. Marivani, E. Tsiligianni, B. Cornelis, and N. Deligiannis, "Multimodal deep unfolding for guided image super-resolution," *IEEE Trans. Image Process.*, vol. 29, pp. 8443–8456, 2020.

[13] X. Deng and P. L. Dragotti, "Deep convolutional neural network for multi-modal image restoration and fusion," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 10, pp. 3333–3348, Oct. 2021.

[14] S. Shaobing, D. L. Donoho, and M. A. Saunders, "Atomic decomposition by basis pursuit," *SIAM Rev.*, vol. 43, no. 1, pp. 129–159, 2001.

[15] J. A. Tropp and A. C. Gilbert, "Signal recovery from random measurements via orthogonal matching pursuit," *IEEE Trans. Inf. Theory*, vol. 53, no. 12, pp. 4655–4666, Dec. 2007.

[16] 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.

[17] K. Engan, S. O. Aase, and J. Hakon Husoy, "Method of optimal directions for frame design," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, Mar. 1999, pp. 2443–2446.

[18] J. Mairal, F. Bach, J. Ponce, and G. Sapiro, "Online dictionary learning for sparse coding," in *Proc. 26th Annu. Int. Conf. Mach. Learn. (ICML)*, 2009, pp. 689–696.

[19] B. Wohlberg, "Efficient algorithms for convolutional sparse representations," *IEEE Trans. Image Process.*, vol. 25, no. 1, pp. 301–315, Jan. 2016.

[20] H. Bristow, A. Eriksson, and S. Lucey, "Fast convolutional sparse coding," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2013, pp. 391–398.

[21] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Found. Trends Mach. Learn.*, vol. 3, no. 1, pp. 1–122, Jul. 2010.

[22] Y. Wang, Q. Yao, J. T. Kwok, and L. M. Ni, "Scalable online convolutional sparse coding," *IEEE Trans. Image Process.*, vol. 27, no. 10, pp. 4850–4859, Oct. 2018.

[23] 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. (ICCV)*, Dec. 2015, pp. 1823–1831.

[24] Y. Liu, X. Chen, R. K. Ward, and Z. J. Wang, "Image fusion with convolutional sparse representation," *IEEE Signal Process. Lett.*, vol. 23, no. 12, pp. 1882–1886, Dec. 2016.

[25] H. Zhang and V. Patel, "Convolutional sparse coding-based image decomposition," in *Proc. Brit. Mach. Vis. Conf.*, 2016, p. 125.

[26] H. Chang *et al.*, "Unsupervised transfer learning via multi-scale convolutional sparse coding for biomedical applications," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 5, pp. 1182–1194, May 2018.

[27] Y. Huang, L. Shao, and A. F. Frangi, "Simultaneous super-resolution and cross-modality synthesis of 3D medical images using weakly-supervised joint convolutional sparse coding," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 6070–6079.

[28] M. Li *et al.*, "Video rain streak removal by multiscale convolutional sparse coding," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 6644–6653.

[29] P. Bao *et al.*, "Convolutional sparse coding for compressed sensing CT reconstruction," *IEEE Trans. Med. Imag.*, vol. 38, no. 11, pp. 2607–2619, Nov. 2019.

[30] S. Wang, L. Zhang, Y. Liang, and Q. Pan, "Semi-coupled dictionary learning with applications to image super-resolution and photo-sketch synthesis," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2012, pp. 2216–2223.

[31] N. Deligiannis, J. F. C. Mota, B. Cornelis, M. R. D. Rodrigues, and I. Daubechies, "Multi-modal dictionary learning for image separation with application in art investigation," *IEEE Trans. Image Process.*, vol. 26, no. 2, pp. 751–764, Feb. 2017.

[32] Y. T. Zhuang, Y. F. Wang, F. Wu, Y. Zhang, and W. M. Lu, "Supervised coupled dictionary learning with group structures for multi-modal retrieval," in *Proc. 27th AAAI Conf. Artif. Intell.*, 2013, pp. 1070–1076.

[33] X.-Y. Jing *et al.*, "Super-resolution person re-identification with semi-coupled low-rank discriminant dictionary learning," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 695–704.

[34] S. Bahrampour, N. M. Nasrabadi, A. Ray, and W. K. Jenkins, "Multimodal task-driven dictionary learning for image classification," *IEEE Trans. Image Process.*, vol. 25, no. 1, pp. 24–38, Jan. 2016.

[35] V. Papyan, Y. Romano, and M. Elad, "Convolutional neural networks analyzed via convolutional sparse coding," *J. Mach. Learn. Res.*, vol. 18, no. 1, pp. 2887–2938, Jan. 2017.

[36] B. Wohlberg, "Convolutional sparse representation of color images," in *Proc. IEEE Southwest Symp. Image Anal. Interpretation (SSIAI)*, Mar. 2016, pp. 57–60.

[37] X. Hu, F. Heide, Q. Dai, and G. Wetzstein, "Convolutional sparse coding for RGB+NIR imaging," *IEEE Trans. Image Process.*, vol. 27, no. 4, pp. 1611–1625, Apr. 2018.

[38] C. A. Barajas-Solano, J.-M. Ramirez, and H. Arguello, "Spectral video compression using convolutional sparse coding," in *Proc. Data Compress. Conf. (DCC)*, Mar. 2020, pp. 253–262.

[39] T. D. La Tour, T. Moreau, M. Jas, and A. Gramfort, "Multivariate convolutional sparse coding for electromagnetic brain signals," in *Proc. Adv. Neural Inf. Process. Syst. (NeurIPS)*, 2018, pp. 3292–3302.

[40] B. Wohlberg. (2016). *SParse Optimization Research Code (SPORCO)*. [Online]. Available: http://purl.org/brendt/software/sporco

[41] (2016). *Sparse Optimization Research Code (SPORCO)*. [Online]. Available: http://purl.org/brendt/software/sporco

[42] C. Garcia-Cardona and B. Wohlberg, "Convolutional dictionary learning: A comparative review and new algorithms," 2017, *arXiv:1709.02893*.

[43] W. W. Hager, "Updating the inverse of a matrix," *SIAM Rev.*, vol. 31, no. 2, pp. 221–239, Jun. 1989.

[44] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: From error visibility to structural similarity," *IEEE Trans. Image Process.*, vol. 13, no. 4, pp. 600–612, Apr. 2004.

[45] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, "Image denoising by sparse 3-D transform-domain collaborative filtering," *IEEE Trans. Image Process.*, vol. 16, no. 8, pp. 2080–2095, Aug. 2007.

[46] P. Song and M. R. D. Rodrigues, "Multimodal image denoising based on coupled dictionary learning," in *Proc. 25th IEEE Int. Conf. Image Process. (ICIP)*, Oct. 2018, pp. 515–519.

[47] X. Guo, Y. Li, J. Ma, and H. Ling, "Mutually guided image filtering," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 3, pp. 694–707, Mar. 2020.

[48] J. Xu, D. Ren, L. Zhang, and D. Zhang, "Patch group based Bayesian learning for blind image denoising," in *Proc. Asian Conf. Comput. Vis. (ACCV)*. Cham, Switzerland: Springer, 2016, pp. 79–95.

[49] Y. Li, J.-B. Huang, N. Ahuja, and M.-H. Yang, "Joint image filtering with deep convolutional networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 8, pp. 1909–1923, Aug. 2019.

[50] G. Chen, F. Zhu, and P. A. Heng, "An efficient statistical method for image noise level estimation," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 477–485.

[51] Y. Liu, S. Liu, and Z. Wang, "A general framework for image fusion based on multi-scale transform and sparse representation," *Inf. Fusion*, vol. 24, pp. 147–164, Jul. 2015.

[52] D. P. Bavirisetti and R. Dhuli, "Fusion of infrared and visible sensor images based on anisotropic diffusion and Karhunen-Loeve transform," *IEEE Sensors J.*, vol. 16, no. 1, pp. 203–209, Jan. 2016.

[53] Y. Liu, X. Chen, R. K. Ward, and Z. J. Wang, "Medical image fusion via convolutional sparsity based morphological component analysis," *IEEE Signal Process. Lett.*, vol. 26, no. 3, pp. 485–489, Mar. 2019.

[54] Z. Zhao, S. Xu, C. Zhang, J. Liu, J. Zhang, and P. Li, "DIDFuse: Deep image decomposition for infrared and visible image fusion," in *Proc. 29th Int. Joint Conf. Artif. Intell.*, Jul. 2020.

[55] H. Xu, J. Ma, J. Jiang, X. Guo, and H. Ling, "U2Fusion: A unified unsupervised image fusion network," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 1, pp. 502–518, Jan. 2022.

[56] G. Qu, D. Zhang, and P. Yan, "Information measure for performance of image fusion," *Electron. Lett.*, vol. 38, no. 7, pp. 313–315, Mar. 2002.

[57] R. Nava, G. Cristóbal, and B. Escalante-Ramrez, "Mutual information improves image fusion quality assessments," *SPIE News Room*, vol. 34, pp. 94–109, Apr. 2007.