Image Deblurring and Super-Resolution by Adaptive Sparse Domain Selection and Adaptive Regularization

Weisheng Dong, Lei Zhang, Member, IEEE, Guangming Shi, Senior Member, IEEE, and Xiaolin Wu, Fellow, IEEE

Abstract—As a powerful statistical image modeling technique, sparse representation has been successfully used in various image restoration applications. The success of sparse representation owes to the development of the $l_1$-norm optimization techniques and the fact that natural images are intrinsically sparse in some domains. The image restoration quality largely depends on whether the employed sparse domain can represent well the underlying image. Considering that the contents can vary significantly across different images or different patches in a single image, we propose to learn various sets of bases from a precollected dataset of example image patches, and then, for a given patch to be processed, one set of bases are adaptively selected to characterize the local sparse domain. We further introduce two adaptive regularization terms into the sparse representation framework. First, a set of autoregressive (AR) models are learned from the dataset of example image patches. The best fitted AR models to a given patch are adaptively selected to regularize the image local structures. Second, the image nonlocal self-similarity is introduced as another regularization term. In addition, the sparsity regularization parameter is adaptively estimated for better image restoration performance. Extensive experiments on image deblurring and super-resolution validate that by using adaptive sparse domain selection and adaptive regularization, the proposed method achieves performance. Extensive experiments on image deblurring and super-resolution validate that by using adaptive sparse domain selection and adaptive regularization, the proposed method achieves

Index Terms—Deblurring, image restoration (IR), regularization, sparse representation, super-resolution.

I. INTRODUCTION

MAGE restoration (IR) aims to reconstruct a high-quality image $\mathbf{x}$ from its degraded measurement $\mathbf{y}$. IR is a typical ill-posed inverse problem [1], and it can be generally modeled as

\[ \mathbf{y} = \mathbf{D} \mathbf{x} + \nu \tag{1} \]

where $\mathbf{x}$ is the unknown image to be estimated, and $\mathbf{H}$ and $\mathbf{D}$ are degrading operators and $\nu$ is additive noise. When $\mathbf{H}$ and $\mathbf{D}$ are identities, the IR problem becomes denoising; when $\mathbf{D}$ is identity and $\mathbf{H}$ is a blurring operator, IR becomes deblurring; when $\mathbf{D}$ is identity and $\mathbf{H}$ is a set of random projections, IR becomes compressed sensing [2]–[4]; when $\mathbf{D}$ is a down-sampling operator and $\mathbf{H}$ is a blurring operator, IR becomes (single-image) super-resolution. As a fundamental problem in image processing, IR has been extensively studied in the past three decades [5]–[20]. In this paper, we focus on deblurring and single-image super-resolution.

Due to the ill-posed nature of IR, the solution to (1) with an $l_2$-norm fidelity constraint, i.e., $\hat{x} = \arg \min_{x} \|y - \mathbf{D}x\|_2^2$, is generally not unique. To find a better solution, prior knowledge of natural images can be used to regularize the IR problem. One of the most commonly used regularization models is the total variation (TV) model [6], [7]:

\[ \hat{x} = \arg \min_{x} \{ \|y - \mathbf{D}x\|_2^2 + \lambda \cdot \|\nabla \mathbf{x}\|_1 \} , \]

where $\|\nabla \mathbf{x}\|_1$ is the $l_1$-norm of the first-order derivative of $\mathbf{x}$ and $\lambda$ is a constant. Since the TV model favors the piece-wise constant image structures, it tends to smooth out the fine details of an image. To better preserve the image edges, many algorithms have been later developed to improve the TV models [17]–[19], [42], [45], [47].

The success of TV regularization validates the importance of good image prior models in solving the IR problems. In wavelet-based image denoising [21], researchers have found that the sparsity of wavelet coefficients can serve as good prior. This reveals the fact that many types of signals, e.g., natural images, can be sparsely represented (or coded) using a dictionary of atoms, such as DCT or wavelet bases, that is, denoting $\Phi$ the dictionary, we have $\mathbf{x} \approx \Phi \alpha$ and most of the coefficients in $\alpha$ are close to zero. With the sparsity prior, the representation of $\mathbf{x}$ over $\Phi$ can be estimated from its observation $\mathbf{y}$ by solving the following $l_0$-minimization problem:

\[ \hat{\alpha} = \arg \min_{\alpha} \{ \|y - \mathbf{D}\Phi\alpha\|_2^2 + \lambda \cdot |\alpha|_0 \} , \]

where the $l_0$-norm counts the number of nonzero coefficients in vector $\alpha$. Once $\hat{\alpha}$ is obtained, $\mathbf{x}$ can then be estimated as $\hat{x} = \Phi \hat{\alpha}$. The $l_0$-minimization is an NP-hard combinatorial search problem, and is usually solved by greedy algorithms [48], [60]. The $l_1$-minimization, as the closest convex function to $l_0$-minimization, is then widely used as an alternative approach to solving the sparse coding problem:

\[ \hat{\alpha} = \arg \min_{\alpha} \{ \|y - \mathbf{D}\Phi\alpha\|_2^2 + \lambda \cdot |\alpha|_1 \} . \]

In addition, recent studies showed that iteratively reweighting the $l_1$-norm sparsity regularization term can lead to better IR results [59]. Sparse representation has been successfully used in various image processing applications [2]–[4], [13], [21]–[25], [32].
A critical issue in sparse representation modeling is the determination of dictionary $\Phi$. Analytically designed dictionaries, such as DCT, wavelet, curvelet, and contourlets, share the advantages of fast implementation; however, they lack the adaptivity to image local structures. Recently, there has been much effort in learning dictionaries from example image patches [13]–[15], [26]–[31], [55], leading to state-of-the-art results in image denoising and reconstruction. Many dictionary learning (DL) methods aim at learning a universal and over-complete dictionary to represent various image structures. However, sparse decomposition over a highly redundant dictionary is potentially unstable and tends to generate visual artifacts [53], [54]. In this paper, we propose an adaptive sparse domain selection (ASDS) scheme for sparse representation. A set of compact subdictionaries is learned from high-quality example image patches. The example image patches are clustered into many clusters. Since each cluster consists of many patches with similar patterns, a compact subdictionary can be learned for each cluster. Particularly, for simplicity, we use the principal component analysis (PCA) technique to learn the subdictionaries. For an image patch to be coded, the best subdictionary that is most relevant to the given patch is selected. Since the given patch can be better represented by the adaptively selected subdictionary, the whole image can be more accurately reconstructed than using a universal dictionary, which will be validated by our experiments.

Apart from the sparsity regularization, other regularization terms can also be introduced to further increase the IR performance. In this paper, we propose to use the piecewise autoregressive (AR) models, which are prelearned from the training dataset, to characterize the local image structures. For each given local patch, one or several AR models can be adaptively selected to regularize the solution space. On the other hand, considering the fact that there are often many repetitive image structures in an image, we introduce a nonlocal (NL) self-similarity constraint served as another regularization term, which is very helpful in preserving edge sharpness and suppressing noise.

After introducing ASDS and adaptive regularizations (AReg) into the sparse representation-based IR framework, we present an efficient iterative shrinkage (IS) algorithm to solve the $l_1$-minimization problem. In addition, we adaptively estimate the image local sparsity to adjust the sparsity regularization parameters. Extensive experiments on image deblurring and super-resolution show that the proposed ASDS-AReg approach can effectively reconstruct the image details, outperforming many state-of-the-art IR methods in terms of both PSNR and visual perception.

The remainder of this paper is organized as follows. Section II introduces the related works. Section III presents the ASDS-based sparse representation. Section IV describes the AReg modeling. Section V summarizes the proposed algorithm. Section VI presents experimental results, and Section VII concludes the paper.

II. RELATED WORKS

It has been found that natural images can be generally coded by structural primitives, e.g., edges and line segments [61], and these primitives are qualitatively similar in form to simple cell receptive fields [62]. In [63], Olshausen et al. proposed to represent a natural image using a small number of basis functions chosen out of an over-complete code set. In recent years, such a sparse coding or sparse representation strategy has been widely studied to solve inverse problems, partially due to the progress of $l_0$-norm and $l_1$-norm minimization techniques [60].

Suppose that $x \in \mathbb{R}^n$ is the target signal to be coded, and $\Phi = [\phi_1, \ldots, \phi_m] \in \mathbb{R}^{n \times n}$ is a given dictionary of atoms (i.e., code set). The sparse coding of $x$ over $\Phi$ is to find a sparse vector $\alpha = [\alpha_1, \ldots, \alpha_m]$ (i.e., most of the coefficients in $\alpha$ are close to zero) such that $x \approx \Phi \alpha$ [49]. If the sparsity is measured as the $l_0$-norm of $\alpha$, which counts the nonzero coefficients in $\alpha$, the sparse coding problem becomes

$$\min_\alpha \| x - \Phi \alpha \|_2^2 \text{ s.t. } |\alpha_i|_0 \leq T,$$

where $T$ is a scalar controlling the sparsity [55]. Alternatively, the sparse vector $\alpha$ can also be found by

$$\hat{\alpha} = \arg \min_\alpha \left\{ \| x - \Phi \alpha \|_2^2 + \lambda \cdot |\alpha|_0 \right\}$$

(2)

where $\lambda$ is a constant. Since the $l_0$-norm is nonconvex, it is often replaced by either the standard $l_1$-norm or the weighted $l_1$-norm to make the optimization problem convex [3], [57], [59], [60].

An important issue of the sparse representation modeling is the choice of dictionary $\Phi$. Much effort has been made in learning a redundant dictionary from a set of example image patches [13]–[15], [26]–[31], [55]. Given a set of training image patches $\mathcal{S} = \{ s_1, \ldots, s_N \} \in \mathbb{R}^{n \times N}$, the goal of dictionary learning (DL) is to jointly optimize the dictionary $\Phi$ and the representation coefficient matrix $\Lambda = [\alpha_1, \ldots, \alpha_N]$ such that

$$s_i \approx \Phi \alpha_i \text{ and } |\alpha_i|_p \leq T,$$

where $p = 0$ or 1. This can be formulated by the following minimization problem:

$$\min_{\Phi, \Lambda} \| \mathcal{S} - \Phi \Lambda \|_F^2 \text{ s.t. } |\alpha_i|_p \leq T; \forall i$$

(3)

where $\| \cdot \|_F$ is the Frobenius norm. The above minimization problem is nonconvex even when $p = 1$. To make it tractable, approximation approaches, including MOD [56] and K-SVD [26], have been proposed to alternatively optimizing $\Phi$ and $\Lambda$, leading to many state-of-the-art results in image processing [14], [15], [31].

Various extensions and variants of the K-SVD algorithm [27], [29]–[31] have been proposed to learn a universal and over-complete dictionary. However, the image contents can vary significantly across images. One may argue that a well-learned over-complete dictionary $\Phi$ can sparsely code all of the possible image structures; nonetheless, for each given image patch, such a “universal” dictionary $\Phi$ is neither optimal nor efficient because many atoms in $\Phi$ are irrelevant to the given local patch. These irrelevant atoms will not only reduce the computational efficiency in sparse coding but also reduce the representation accuracy.

Regularization has been used in IR for a long time to incorporate the image prior information. The widely used TV regularizations lack flexibilities in characterizing the local image structures and often generate over-smoothed results. As a classic method, the autoregressive (AR) modeling has been successfully used in image compression [33] and interpolation [34],
Recently, the AR model was used for adaptive regularization in compressive image recovery [40]:
\[
\min_{\mathbf{z}} \sum_{i} \| x_i - \mathbf{X}_k \mathbf{a}_k \|^2_2
\]
s.t. \( \mathbf{y} = \mathbf{A}\mathbf{z} \), where \( \mathbf{X}_k \) is the vector containing the neighboring pixels of pixel \( x_i \) within the support of the AR model, and \( \mathbf{a}_k \) is the AR parameter vector. In [40], the AR models are locally computed from an initially recovered image, and they perform much better than the TV regularization in reconstructing the edge structures. However, the AR models estimated from the initially recovered image may not be robust and tend to produce the “ghost” visual artifacts. In this paper, we will propose a learning-based adaptive regularization, where the AR models are learned from high-quality training images, to increase the AR modeling accuracy.

In recent years, the nonlocal (NL) methods have led to promising results in various IR tasks, especially in image denoising [15], [36], [39]. The mathematical framework of NL means filtering was well established by Buades et al. [36]. The idea of NL methods is very simple: the patches that have similar patterns can be spatially far from each other, and thus we can collect them in the whole image. This NL self-similarity prior was later employed in image deblurring [8], [20] and super-resolution [41]. In [15], the NL self-similarity prior was combined with the sparse representation modeling, where the similar image patches are simultaneously coded to improve the robustness of inverse reconstruction. In this work, we will also introduce an NL self-similarity regularization term into our proposed IR framework.

### III. SPARSE REPRESENTATION WITH ADAPTIVE SPARSE DOMAIN SELECTION (ASDS)

Here, we propose an ASDS scheme, which learns a series of compact subdictionaries and assigns adaptively each local patch a subdictionary as the sparse domain. With ASDS, a weighted \( l_1 \)-norm sparse representation model will be proposed for IR tasks. Suppose that \( \{ \Phi_k \} \), \( k = 1, 2, \ldots, K \), is a set of \( K \) orthonormal subdictionaries. Let \( \mathbf{x} \) be an image vector, and \( \mathbf{x}_i = \mathbf{R}_i \mathbf{x} \), \( i = 1, 2, \ldots, N \), be the \( i \)th patch (size: \( \sqrt{m} \times \sqrt{m} \)) vector of \( \mathbf{x} \), where \( \mathbf{R}_i \) is a matrix extracting patch \( \mathbf{x}_i \) from \( \mathbf{x} \). For patch \( \mathbf{x}_i \), suppose that a subdictionary \( \Phi_{k_i} \) is selected for it. Then, \( \mathbf{x}_i \) can be approximated as \( \mathbf{x}_i = \Phi_{k_i} \mathbf{a}_i \), \( \| \mathbf{a}_i \|_1 \leq T \), via sparse coding. The whole image \( \mathbf{x} \) can be reconstructed by averaging all of the reconstructed patches \( \hat{x}_i \), which can be mathematically written as [22]

\[
\hat{x} = \left( \sum_{i=1}^{N} R_i^T R_i \right)^{-1} \left( \sum_{i=1}^{N} R_i^T \Phi_{k_i} \mathbf{a}_i \right). \tag{4}
\]

In (4), the matrix to be inverted is a diagonal matrix, and hence the calculation of (4) can be done in a pixel-by-pixel manner [22]. Obviously, the image patches can be overlapped to better suppress noise [15], [22] and block artifacts. For the convenience of expression, we define the following operator “\( \Psi \)”: \( \hat{x} = \Phi \circ \mathbf{a} \triangleq \left( \sum_{i=1}^{N} R_i^T R_i \right)^{-1} \left( \sum_{i=1}^{N} R_i^T \Phi_{k_i} \mathbf{a}_i \right) \). The remaining problem is how to learn a subdictionary \( \Phi_{k_i} \) from the cluster \( S_{k_i} \) such that all the elements in \( S_{k_i} \) are faithfully represented by \( \Phi_{k_i} \). Meanwhile, we hope that the representation of \( S_{k_i} \) over \( \Phi_{k_i} \) is as sparse as possible. The design

\[
\hat{x} = \Psi \circ \mathbf{a} \triangleq \left( \sum_{i=1}^{N} R_i^T R_i \right)^{-1} \left( \sum_{i=1}^{N} R_i^T \Phi_{k_i} \mathbf{a}_i \right) \tag{5}
\]

where \( \Phi \) is the concatenation of all subdictionaries \( \{ \Phi_k \} \) and \( \mathbf{a} \) is the concatenation of all \( \mathbf{a}_k \).

Let \( \mathbf{y} = \mathbf{D} \mathbf{h} + \mathbf{v} \) be the observed degraded image, our goal is to recover the original image \( \mathbf{x} \) from \( \mathbf{y} \). With ASDS and the definition in (5), the IR problem can be formulated as follows:

\[
\hat{\mathbf{a}} = \arg \min_{\mathbf{a}} \left\{ \| \mathbf{y} - \mathbf{D} \Phi \circ \mathbf{a} \|_2^2 + \lambda \| \mathbf{a} \|_1 \right\}. \tag{6}
\]

Clearly, one key procedure in the proposed ASDS scheme is the determination of \( \Phi_{k_i} \) for each local patch. To facilitate the sparsity-based IR, we propose to learn offline the subdictionaries \( \{ \Phi_k \} \), and select online from \( \{ \Phi_k \} \) the best fitted subdictionary to each patch \( \mathbf{x}_i \).

#### A. Learning the Subdictionaries

In order to learn a series of subdictionaries to code the various local image structures, we need to first construct a dataset of local image patches for training. To this end, we collected a set of high-quality natural images and cropped from them a rich amount of image patches with size \( \sqrt{m} \times \sqrt{m} \). A cropped image patch, denoted by \( \mathbf{s}_k \), will be involved in DL if its intensity variance \( \text{Var}(\mathbf{s}_k) \) is greater than a threshold \( \Delta \), i.e., \( \text{Var}(\mathbf{s}_k) > \Delta \). This patch selection criterion is to exclude the smooth patches from training and guarantee that only the meaningful patches with a certain amount of edge structures are involved in DL.

Suppose that \( M \) image patches \( \mathbf{S} = [\mathbf{s}_1, \mathbf{s}_2, \ldots, \mathbf{s}_M] \) are selected. We aim to learn \( K \) compact subdictionaries \( \{ \Phi_k \} \) from \( \mathbf{S} \) so that, for each given local image patch, the most suitable subdictionary can be selected. To this end, we cluster the dataset \( \mathbf{S} \) into \( K \) clusters, and learn a subdictionary from each of the \( K \) clusters. Apparently, the \( K \) clusters are expected to represent the \( K \) distinctive patterns in \( \mathbf{S} \). To generate perceptually meaningful clusters, we perform the clustering in a feature space. In the hundreds of thousands patches cropped from the training images, many patches are approximately the rotated version of the others. Hence, we do not need to explicitly make the training dataset invariant to rotation because it is naturally (nearly) rotation invariant. Considering the fact that human visual system is sensitive to image edges, which convey most of the semantic information of an image, we use the high-pass filtering output of each patch as the feature for clustering. It allows us to focus on the edges and structures of image patches without taking into account the pixel intensities and helps to increase the accuracy of clustering. The high-pass filtering is often used in low-level statistical learning tasks to enhance the meaningful features [50].

Denote by \( \mathbf{S}_{k_i} = [\mathbf{s}_{k_i,1}, \mathbf{s}_{k_i,2}, \ldots, \mathbf{s}_{k_i,K}] \) the high-pass filtered dataset of \( \mathbf{S} \). We adopt the \( K \)-means algorithm to partition \( \mathbf{S}_{k_i} \) into \( K \) clusters \( \{ C_{k,j} \} \) and denote by \( \mu_{k,j} \) the centroid of cluster \( C_{k,j} \). Once \( \mathbf{S}_{k_i} \) is partitioned, dataset \( \mathbf{S} \) can then be clustered into \( K \) subsets \( \mathbf{S}_{k_i} \), \( k = 1, 2, \ldots, K \), and \( \mathbf{S}_{k_i} \) is a matrix of dimension \( n \times m_{k_i} \), where \( m_{k_i} \) denotes the number of samples in \( \mathbf{S}_{k_i} \).

Now the remaining problem is how to learn a subdictionary \( \Phi_{k_i} \) from the cluster \( \mathbf{S}_{k_i} \) such that all the elements in \( \mathbf{S}_{k_i} \) can be faithfully represented by \( \Phi_{k_i} \). Meanwhile, we hope that the representation of \( \mathbf{S}_{k_i} \) over \( \Phi_{k_i} \) is as sparse as possible. The design

\[
\hat{x} = \Psi \circ \mathbf{a} \triangleq \left( \sum_{i=1}^{N} R_i^T R_i \right)^{-1} \left( \sum_{i=1}^{N} R_i^T \Phi_{k_i} \mathbf{a}_i \right) \tag{5}
\]
of $\Phi_k$ can be intuitively formulated by the following objective function:

$$
(\hat{\Phi}_k, \hat{\Lambda}_k) = \arg \min_{\Phi_k, \Lambda_k} \left\{ \|S_k - \Phi_k \Lambda_k\|^2_F + \lambda \|\Lambda_k\|_1 \right\} \quad (7)
$$

where $\Lambda_k$ is the representation coefficient matrix of $S_k$ over $\Phi_k$. Equation (7) is a joint optimization problem of $\Phi_k$ and $\Lambda_k$, and it can be solved by alternatively optimizing $\Phi_k$ and $\Lambda_k$, like in the K-SVD algorithm [26].

However, we do not directly use (7) to learn the subdictionary $\Phi_k$ based on the following considerations. First, the $l_2 - l_1$ joint minimization in (7) requires much computational cost. Second, and more important, by using the objective function in (7), we often assume that the dictionary $\Phi_k$ is over-complete. Nonetheless, here $S_k$ is a subdataset after $K$-means clustering, which implies that, not only is the number of elements in $S_k$ limited, but also these elements tend to have similar patterns. Therefore, it is not necessary to learn an over-complete dictionary $\Phi_k$ from $S_k$. In addition, a compact dictionary will decrease much the computational cost of the sparse coding of a given image patch. With the above considerations, we propose to learn a compact dictionary while trying to approximate (7). The PCA is a good solution to this end.

PCA is a classical signal de-correlation and dimensionality reduction technique that is widely used in pattern recognition and statistical signal processing [37]. In [38] and [39], PCA has been successfully used in spatially adaptive image denoising by computing the local PCA transform of each image patch. In this paper, we apply PCA to each subdataset $S_k$ to compute the principal components, from which the dictionary $\Phi_k$ is constructed. Denote by $\Omega_k$ the co-variance matrix of dataset $S_k$. By applying PCA to $\Omega_k$, an orthogonal transformation matrix $P_k$ can be obtained. If we set $P_k$ as the dictionary and let $Z_k = P_k^T S_k$, we will then have $\|S_k - P_k Z_k\|^2_F = \|S_k - P_k P_k^T S_k\|^2_F = 0$. In other words, the approximation term in (7) will be exactly zero, yet the corresponding sparsity regularization term $\|\Lambda_k\|_1$ will have a certain amount because all the representation coefficients in $Z_k$ are preserved.

To make a better balance between the $l_1$-norm regularization term and $l_2$-norm approximation term in (7), we only extract the first $r$ most important eigenvectors in $P_k$ to form a dictionary $\Phi_r$, i.e., $\Phi_r = [p_1, p_2, \ldots, p_r]$. Let $\Lambda_r = \Phi_r^T S_k$. Clearly, since not all of the eigenvectors are used to form $\Phi_r$, the reconstruction error $\|S_k - \Phi_r \Lambda_r\|^2_F$ in (7) will increase with the decrease of $r$. However, the term $\|\Lambda_r\|_1$ will decrease. Therefore, the optimal value of $r$, denoted by $r_0$, can be determined by

$$
r_0 = \arg \min_r \left\{ \|S_k - \Phi_r \Lambda_r\|^2_F + \lambda \|\Lambda_r\|_1 \right\}. \quad (8)
$$

Finally, the subdictionary learned from subdataset $S_k$ is $\Phi_k = [p_1, p_2, \ldots, p_{r_0}]$.

Applying the above procedures to all of the $K$ subdatasets $S_k$, we could get $K$ subdictionaries $\Phi_k$, which will be used in the ASDS process of each given image patch. In Fig. 1, we show some example subdictionaries learned from a training dataset. The left column shows the centroids of some subdatasets after $K$-means clustering, and the right eight columns show the first eight atoms in the subdictionaries learned from the corresponding subdatasets.

![Examples of learned subdictionaries. The left column shows the centroids of some subdatasets after K-means clustering, and the right eight columns show the first eight atoms of the learned subdictionaries from the corresponding subdatasets.](image)

B. Adaptive Selection of the Subdictionary

In the previous subsection, we have learned a dictionary $\Phi_k$ for each subset $S_k$. Meanwhile, we have computed the centroid $\mu_k$ of each cluster $C_k$ associated with $S_k$. Therefore, we have $K$ pairs $\{\Phi_k, \mu_k\}$, with which the ASDS of each given image patch can be accomplished.

In the proposed sparsity-based IR scheme, we assign adaptively a subdictionary to each local patch of $x$, spanning the adaptive sparse domain. Since $x$ is unknown beforehand, we need to have an initial estimation of it. The initial estimation of $x$ can be accomplished by taking wavelet bases as the dictionary and then solving (6) with the iterated shrinkage algorithm in [10]. Denote by $\hat{x}$ the estimate of $x$ and denote by $\hat{z}_i$ a local patch of $\hat{x}$. Recall that we have the centroid $\mu_k$ of each cluster available, and hence we could select the best fitted subdictionary to $\hat{z}_i$ by comparing the high-pass filtered patch of $\hat{z}_i$, denoted by $\hat{z}_i^h$, to the centroid $\mu_k$. For example, we can select the dictionary for $\hat{z}_i$ based on the minimum distance between $\hat{z}_i^h$ and $\mu_k$, i.e.,

$$
k_i = \arg \min_k \|\hat{z}_i^h - \mu_k\|_2. \quad (9)
$$

However, directly calculating the distance between $\hat{z}_i^h$ and $\mu_k$ may not be robust enough because the initial estimate $\hat{x}$ can be noisy. Here, we propose to determine the subdictionary in the subspace of $\mu_k$. Let $U = [\mu_1, \mu_2, \ldots, \mu_K]$ be the matrix containing all the centroids. By applying SVD to the co-variance matrix of $U$, we can obtain the PCA transformation matrix of $U$. Let $\Phi$ be the projection matrix composed by the first several most significant eigenvectors. We compute the distance between $\hat{z}_i^h$ and $\mu_k$ in the subspace spanned by $\Phi$, as

$$
k_i = \arg \min_k \|\Phi \hat{z}_i^h - \Phi \mu_k\|_2. \quad (10)
$$

Compared with (9), (10) can increase the robustness of adaptive dictionary selection.
By using (10), the kth subdictionary \( \Phi_k \) will be selected and assigned to patch \( \hat{x}_i \). Then, we can update the estimation of \( \alpha \) by minimizing (6) and letting \( \hat{x} = \Phi \circ \hat{\alpha} \). With the updated estimate \( \hat{\alpha} \), the ASDS of \( x \) can be consequently updated. Such a process is iteratively implemented until the estimation \( \hat{\alpha} \) converges.

C. Adaptively Reweighted Sparsity Regularization

In (6), the parameter \( \lambda \) is a constant to weight the \( l_1 \)-norm sparsity regularization term \( ||\alpha||_1 \). In [59] Candès et al. showed that the reweighted \( l_1 \)-norm sparsity can more closely resemble the \( l_0 \)-norm sparsity than using a constant weight and consequently improve the reconstruction of sparse signals. Here, we propose a new method to estimate adaptively the image local sparsity and then reweight the \( l_1 \)-norm sparsity in the ASDS scheme.

The reweighted \( l_1 \)-norm sparsity regularized minimization with ASDS can be formulated as follows:

\[
\hat{\alpha} = \arg\min_{\alpha} \left\{ ||y - D\Phi \circ \alpha||_2 + \sum_{i=1}^{N} \sum_{j=1}^{n} \lambda_{i,j}(\alpha_{i,j}) \right\}
\]

(11)

where \( \alpha_{i,j} \) is the coefficient associated with the \( j \)th atom of \( \Phi_{ki} \) and \( \lambda_{i,j} \) is the weight assigned to \( \alpha_{i,j} \). In [59], \( \lambda_{i,j} \) is empirically computed as \( \lambda_{i,j} = 1/\langle |\alpha_{i,j}| + \varepsilon \rangle \), where \( \varepsilon \) is the estimate of \( \sigma_{i,j} \) and \( \varepsilon \) is a small constant. Here, we propose a more robust method for computing \( \lambda_{i,j} \) by formulating the sparsity estimation as a maximum a posteriori (MAP) estimation problem. Under the Bayesian framework, with the observation \( y \) the MAP estimation of \( \alpha \) is given by

\[
\hat{\alpha} = \arg\max_{\alpha} \{ \log P(\alpha | y) \} = \arg\min_{\alpha} \left\{ -\log P(y | \alpha) - \log P(\alpha) \right\}.
\]

(12)

By assuming \( y \) is contaminated with additive Gaussian white noises of standard deviation \( \sigma_n \), we have

\[
P(y | \alpha) = \frac{1}{\sigma_n \sqrt{2\pi}} \exp\left( -\frac{1}{2\sigma_n^2} ||y - D\Phi \circ \alpha||_2^2 \right).
\]

(13)

The prior distribution \( P(\alpha) \) is often characterized by an i.i.d. zero-mean Laplacian probability model

\[
P(\alpha) = \prod_{i=1}^{N} \prod_{j=1}^{n} \frac{1}{\sqrt{2\sigma_{i,j}}} \exp\left( -\frac{\sqrt{2}}{\sigma_{i,j}} |\alpha_{i,j}| \right)
\]

(14)

where \( \sigma_{i,j} \) is the standard deviation of \( \alpha_{i,j} \). By plugging \( P(y | \alpha) \) and \( P(\alpha) \) into (12), we could readily derive the desired weight in (11) as \( \lambda_{i,j} = 2\sqrt{2} \sigma_{i,j}^2 / \sigma_{i,j} \). For numerical stability, we compute the weights by

\[
\lambda_{i,j} = \frac{2\sqrt{2} \sigma_{i,j}^2}{\sigma_{i,j} + \varepsilon}
\]

(15)

where \( \sigma_{i,j} \) is an estimate of \( \sigma_{i,j} \) and \( \varepsilon \) is a small constant.

Now, let us discuss how to estimate \( \sigma_{i,j} \). Denote by \( \hat{x}_i \) the estimate of \( x_i \), and by \( \hat{x}_i^l \), \( l = 1, 2, \ldots, L \), the nonlocal similar patches to \( \hat{x}_i \). The (determination of nonlocal similar patches to \( \hat{x}_i \) will be described in Section IV-C.) The representation coefficients of these similar patches over the selected subdictionary \( \Phi_k \) is \( \hat{\alpha}_i^l = \Phi_k^l \hat{x}_i^l \). Then, we can estimate \( \sigma_{i,j} \) by calculating the standard deviation of each element \( \hat{\alpha}_{i,j}^l \) in \( \hat{\alpha}_i^l \). Compared with the reweighting method in [59], the proposed adaptive reweighting method is more robust because it exploits the image nonlocal redundancy information. Based on our experimental experience, it could lead to about 0.2 dB improvement in average over the reweighting method in [59] for deblurring and super-resolution under the proposed ASDS framework. The detailed algorithm to solve the reweighted \( l_1 \)-norm sparsity regularized minimization in (11) will be presented in Section V.

IV. SPATIALLY ADAPTIVE REGULARIZATION

In Section III, we proposed to select adaptively a subdictionary to code the given image patch. The proposed ASDS-based IR method can be further improved by introducing two types of adaptive regularization (AReg) terms. A local area in a natural image can be viewed as a stationary process, which can be well modeled by the autoregressive (AR) models. Here, we propose to learn a set of AR models from the clustered high quality training image patches, and adaptively select one AR model to regularize the input image patch. Besides the AR models, which exploit the image local correlation, we propose to use the nonlocal similarity constraint as a complementary AReg term to the local AR models. With the fact that there are often many repetitive image structures in natural images, the image nonlocal redundancies can be very helpful in image enhancement.

A. Training the AR Models

Recall that, in Section III, we have partitioned the whole training dataset into \( K \) subdatasets \( S_k \). For each \( S_k \), an AR model can be trained using all of the sample patches inside it. Here we let the support of the AR model be a square window, and the AR model aims to predict the central pixel of the window by using the neighboring pixels. Considering that determining the best order of the AR model is not trivial, and a high order AR model may cause data over-fitting, in our experiments a 3 \( \times \) 3 window (i.e., AR model of order 8) is used. The vector of AR model parameters, denoted by \( \alpha_k \), of the \( k \)th subdataset \( S_k \), can be easily computed by solving the following least square problem:

\[
\alpha_k = \arg\min_{\alpha_k} \sum_{s_i \in S_k} (s_i - \alpha^T S_k q_i)^2
\]

(16)

where \( s_i \) is the central pixel of image patch \( s_i \) and \( q_i \) is the vector that consists of the neighboring pixels of \( s_i \) within the support of the AR model. By applying the AR model training process to each subdataset, we can obtain a set of AR models \( \{\alpha_1, \alpha_2, \ldots, \alpha_K\} \) that will be used for adaptive regularization.

B. Adaptive Selection of the AR Model for Regularization

The adaptive selection of the AR model for each patch \( x_i \) is the same as the selection of a subdictionary for \( x_i \) described in Section III-B. With an estimation \( \hat{x}_i \) of \( x_i \), we compute its high-pass Gaussian filtering output \( \hat{x}_i^h \). Let \( k_i = \arg\min_k ||\Phi_k \hat{x}_i^h - \Phi_k \hat{x}_i||_2 \), and then the \( k_i \)th AR model \( \alpha_{k_i} \) will be assigned to patch \( x_i \). Denote by \( X_i \) the central pixel of patch \( x_i \) and by \( \chi_i \) the vector containing the neighboring pixels of \( x_i \) within patch \( x_i \). We can expect that the prediction
error of \( x_i \) using \( a_{i,k} \) and \( \chi_i \) should be small, i.e., \( \| x_i - a_{i,k}^T \chi_i \|_2^2 \) should be minimized. By incorporating this constraint into the ASDS-based sparse representation model in (11), we have a lifted objective function as follows:

\[
\hat{\alpha} = \arg \min_{\alpha} \left\{ \| y - DH \Phi \circ \alpha \|_2^2 + \sum_{i=1}^{n} \sum_{j=1}^{N} \lambda_{i,j} |a_{i,j}| \right. \\
\left. + \gamma \cdot \sum_{x_i \in x} \| x_i - a_{i,k}^T \chi_i \|_2^2 \right\} 
\] (17)

where \( \gamma \) is a constant balancing the contribution of the AR regularization term. For the convenience of expression, we write the third term \( \sum_{x_i \in x} \| x_i - a_{i,k}^T \chi_i \|_2^2 \sim \| (I - A)x \|_2^2 \), where \( I \) is the identity matrix and

\[
A(i,j) = \begin{cases} a_{i,j} & \text{if } x_j \text{ is an element of } x_i; a_i \in a_{i,k} \\
0 & \text{otherwise.} \end{cases}
\]

Then, (17) can be rewritten as

\[
\hat{\alpha} = \arg \min_{\alpha} \left\{ \| y - DH \Phi \circ \alpha \|_2^2 + \sum_{i=1}^{n} \sum_{j=1}^{N} \lambda_{i,j} |a_{i,j}| \right. \\
\left. + \gamma \cdot \| (I - A)x \|_2^2 \right\}. 
\] (18)

C. Adaptive Regularization by Nonlocal Similarity

The AR model-based AReg exploits the local statistics in each image patch. On the other hand, there are often many repetitive patterns throughout a natural image. Such nonlocal redundancy is very helpful to improve the quality of reconstructed images. As a complementary ARReg term to AR models, we further introduce a nonlocal similarity regularization term into the sparsity-based IR framework.

For each image patch \( x_i \), we search for the similar patches to it in the whole image \( x \) (in practice, in a sufficiently large area around \( x_i \)). A patch \( x_i^j \) is selected as a similar patch to \( x_i \) if \( c_{i,j} = \| x_i - x_i^j \|_2^2 \leq t \), where \( t \) is a preset threshold, and \( x_i^j \) and \( x_i \) are the current estimates of \( x_i \) and \( x_i^j \), respectively, or we can select the patch \( x_i^j \) if it is within the first \( L \) (\( L = 10 \) in our experiments) closest patches to \( x_i \). Let \( x_i \) be the central pixel of patch \( x_i \), and \( x_i^j \) be the central pixel of patch \( x_i^j \). Then, we can use the weighted average of \( x_i^j \), i.e., \( \sum_{j=1}^{L} b_{i,j}^T x_i^j \), to predict \( x_i \), and the weight \( b_{i,j}^T \) assigned to \( x_i^j \) is set as \( b_{i,j}^T = \exp(-c_{i,j}^t/h_c) \), where \( h_c \) is a controlling factor of the weight and \( c_{i,j} = \sum_{j=1}^{L} \exp(-c_{i,j}^t/h_c) \) is the normalization factor. Considering that there is much nonlocal redundancy in natural images, we expect that the prediction error \( \| x_i - \sum_{j=1}^{L} b_{i,j}^T x_i^j \|_2^2 \) should be small. Let \( b_i \) be the column vector containing all the weights \( b_{i,j}^T \) and \( \beta_i \) be the column vector containing all \( x_i^j \). By incorporating the nonlocal similarity regularization term into the ASDS-based sparse representation in (11), we have

\[
\hat{\alpha} = \arg \min_{\alpha} \left\{ \| y - DH \Phi \circ \alpha \|_2^2 + \sum_{i=1}^{n} \sum_{j=1}^{N} \lambda_{i,j} |a_{i,j}| \right. \\
\left. + \gamma \cdot \sum_{x_i \in x} \| x_i - b_{i}^T \beta_i \|_2^2 \right\} 
\] (19)

where \( \eta \) is a constant balancing the contribution of nonlocal regularization. Equation (19) can be rewritten as

\[
\hat{\alpha} = \arg \min_{\alpha} \left\{ \| y - DH \Phi \circ \alpha \|_2^2 + \sum_{i=1}^{n} \sum_{j=1}^{N} \lambda_{i,j} |a_{i,j}| \right. \\
\left. + \gamma \cdot \| (I - B)\Phi \circ \alpha \|_2^2 \right\} 
\] (20)

where \( I \) is the identity matrix and

\[
B(i,j) = \begin{cases} b_{i,j} & \text{if } x_i^j \text{ is an element of } \beta_i; b_i \in b_i \\
0 & \text{otherwise.} \end{cases}
\]

V. SUMMARY OF THE ALGORITHM

By incorporating both the local AR regularization and the nonlocal similarity regularization into the ASDS-based sparse representation in (11), we have the following ASDS-ARReg-based sparse representation to solve the IR problem:

\[
\hat{\alpha} = \arg \min_{\alpha} \left\{ \| y - DH \Phi \circ \alpha \|_2^2 + \gamma \cdot \| (I - A)\Phi \circ \alpha \|_2^2 \right. \\
\left. + \eta \cdot \| (I - B)\Phi \circ \alpha \|_2^2 + \sum_{i=1}^{n} \sum_{j=1}^{N} \lambda_{i,j} |a_{i,j}| \right\}, 
\] (21)

In (21), the first \( l_2 \)-norm term is the fidelity term, guaranteeing that the solution \( \hat{x} = \Phi \circ \hat{\alpha} \) can well fit the observation \( y \) after degradation by operators \( H \) and \( D \); the second \( l_2 \)-norm term is the local AR model-based adaptive regularization term, requiring that the estimated image is locally stationary; the third \( l_2 \)-norm term is the nonlocal regularization term, which uses the nonlocal redundancy to enhance each local patch; and the last weighted \( l_1 \)-norm term is the sparsity penalty term, requiring that the estimated image should be sparse in the adaptively selected domain. Equation (21) can be rewritten as

\[
\hat{\alpha} = \arg \min_{\alpha} \left\{ \| y - DH \Phi \circ \alpha \|_2^2 + \gamma \cdot \| (I - A)\Phi \circ \alpha \|_2^2 \right. \\
\left. + \eta \cdot \| (I - B)\Phi \circ \alpha \|_2^2 + \sum_{i=1}^{n} \sum_{j=1}^{N} \lambda_{i,j} |a_{i,j}| \right\}. 
\] (22)

By letting

\[
\hat{y} = \begin{bmatrix} y \\ 0 \end{bmatrix}, \quad K = \begin{bmatrix} DH \\ \gamma \cdot (I - A) \end{bmatrix}, \quad \Phi \circ \alpha = \begin{bmatrix} \Phi \circ \alpha \end{bmatrix}
\]

(23)

(22) can be rewritten as

\[
\hat{\alpha} = \arg \min_{\alpha} \left\{ \| \hat{y} - K \Phi \circ \alpha \|_2^2 + \sum_{i=1}^{n} \sum_{j=1}^{N} \lambda_{i,j} |a_{i,j}| \right\}. 
\] (24)

This is a reweighted \( l_1 \)-minimization problem, which can be effectively solved by the iterative shrinkage algorithm [10]. We outline the iterative shrinkage algorithm for optimizing (24) in Algorithm 1.
Algorithm 1

1) Initialization:
   a) By taking the wavelet domain as the sparse domain, we can compute an initial estimate, denoted by \( \hat{x} \), of \( x \) by using the iterated wavelet shrinkage algorithm \([10]\);
   b) With the initial estimate \( \hat{x} \), we select the subdictionary \( \Phi_{ki} \) and the AR model \( a_i \) using (10), and calculate the nonlocal weight \( b_i \) for each local patch \( \hat{x}_i \);
   c) Initialize \( A \) and \( B \) with the selected AR models and the nonlocal weights;
   d) Preset \( \gamma, \eta, P, e \) and the maximal iteration number, denoted by \( Max_{Iter} \);
   e) Set \( k = 0 \).

2) Iterate on \( k \) until \( \|\hat{x}^{(k)} - \hat{x}^{(k+1)}\|^2 / N \leq e \) or \( k \geq Max_{Iter} \) is satisfied.
   a) \( \hat{x}^{(k+1/2)} = \hat{x}^{(k)} + K^T (y - K \hat{x}^{(k)}) = \hat{x}^{(k)} + (U y - U \hat{x}^{(k)}) - D \hat{x}^{(k)} \), where \( U = (DH)^T D \) and \( V = \gamma^2 (I - A)^T (I - A) + \eta^2 (I - B)^T (I - B) \);
   b) Compute \( \alpha^{(k+1/2)} = \{ \Phi_{ki} R_{ki} \hat{x}^{(k+1/2)} \} \)
   c) \( \alpha^{(k+1)} = \text{soft}(\alpha^{(k+1/2)} / \tau_{ij}) \), where \( \text{soft}(\cdot, \tau_{ij}) \) is a soft thresholding function with threshold \( \tau_{ij} \);
   d) Compute \( \hat{x}^{(k+1)} = \Phi \hat{a}^{(k+1)} \) using (5), which can be calculated by first reconstructing each image patch with \( \hat{x}_i = \Phi_{ki} \hat{a}_i^{(k+1)} \) and then averaging all the reconstructed image patches;
   e) If \( mod(k, P) = 0 \), update the adaptive sparse domain of \( x \) and the matrices \( A \) and \( B \) using the improved estimate \( \hat{x}^{(k+1)} \).

In Algorithm 1, \( e \) is a prespecified scalar controlling the convergence of the iterative process, and \( Max_{Iter} \) is the allowed maximum number of iterations. The thresholds \( \tau_{ij} \) are locally computed as \( \tau_{ij} = \lambda_{ij} / \sigma_{ij} \) \([10]\), where \( \lambda_{ij} \) are calculated by (15) and \( \sigma_{ij} \) is chosen such that \( r > \| (K \Phi)^T K \Phi \|_2 \). Since the dictionary \( \Phi_{ki} \) varies across the image, the optimal determination of \( r \) for each local patch is difficult. Here, we empirically set \( r = 4.7 \) for all of the patches. \( P \) is a preset integer, and we only update the subdictionaries \( \Phi_{ki} \), the AR models \( a_i \) and the weights \( b_i \) in every \( P \) iterations to save computational cost. With the updated \( a_i \) and \( b_i \) and \( A \) and \( B \) can be updated, and then the matrix \( V \) can be updated.

VI. EXPERIMENTAL RESULTS

A. Training Datasets

Although image contents can vary a lot from image to image, it has been found that the micro-structures of images can be represented by a small number of structural primitives (e.g., edges, line segments and other elementary features), and these primitives are qualitatively similar in form to simple cell receptive fields \([61]–[63]\). The human visual system employs a sparse coding strategy to represent images, i.e., coding a natural image using a small number of basis functions chosen out of an over-complete code set. Therefore, using the many patches extracted from several training images which are rich in edges and textures, we are able to train the dictionaries which can represent well the natural images. To illustrate the robustness of the proposed method to the training dataset, we use two different sets of training images in the experiments, each set having five high-quality images as shown in Fig. 2. We can see that these two sets of training images are very different in contents. We use \( Var(s_\ast) > \Delta \) with \( \Delta = 16 \) to exclude the smooth image patches, and a total amount of 727615 patches of size \( 7 \times 7 \) are randomly cropped from each set of training images. (Please refer to Section VI-E for the discussion of patch size selection.)

As a clustering-based method, an important issue is the selection of the number of classes. However, the optimal selection of this number is a nontrivial task, which is subject to the bias and variance tradeoff. If the number of classes is too small, the boundaries between classes will be smoothed out and thus the distinctiveness of the learned subdictionaries and AR models is decreased. On the other hand, too large a number of classes will make the learned subdictionaries and AR models less representative and less reliable. Based on the above considerations and our experimental experience, we propose the following simple method to find a good number of classes: we first partition the training dataset into 200 clusters, and merge those classes that contain very few image patches (i.e., less than 300 patches) to their nearest neighboring classes. More discussions and experiments on the selection of the number of classes will be made in Section VI-E.

B. Experimental Settings

In the experiments of deblurring, two types of blur kernels, a Gaussian kernel of standard deviation 3 and a \( 9 \times 9 \) uniform kernel, were used to simulate blurred images. Additive Gaussian white noises with standard deviations \( \sqrt{2} \) and 2 were then added to the blurred images, respectively. We compare the proposed methods with five recently proposed image deblurring methods: the iterated wavelet shrinkage method \([10]\), the constrained TV deblurring method \([42]\), the spatially weighted TV deblurring method \([45]\), the \( l_0 \)-norm sparsity based deblurring method \([46]\), and the BM3D deblurring method \([58]\). In the proposed ASDS-AReg Algorithm 1, we empirically set \( \gamma = 0.00775 \), \( \eta = 0.1414 \), and \( \tau_{ij} = \lambda_{ij} / 4.7 \), where \( \lambda_{ij} \) is adaptively computed by (15).

In the experiments of super-resolution, the degraded LR images were generated by first applying a truncated 7 \( \times \) 7 Gaussian kernel of standard deviation 1.6 to the original image and then down-sampling by a factor of 3. We compare the proposed method with four state-of-the-art methods: the iterated wavelet shrinkage method \([10]\), the TV-regularization based method \([47]\), the Softcuits method \([43]\), and the sparse representation based method \([25]\). Since the method in \([25]\) does not handle the blurring of LR images, for fair comparisons we use the iterative back-projection method \([16]\) to deblur the HR images produced by \([25]\). In the proposed ASDS-AReg based super-resolution, the parameters are set as follows. For the noiseless LR images, we empirically set \( \gamma = 0.0094 \), \( \eta = 0.2 \) and \( \tau_{ij} = 0.18 / \sigma_{ij} \), where \( \sigma_{ij} \) is the estimated standard

\[ \text{We thank the authors of \([42]\), \([43]\), \([45]\), \([46]\), \([58]\) and \([25]\) for providing their source codes, executable programs, or experimental results.} \]
deviation of $\alpha_{i,j}$. For the noisy LR images, we empirically set $\gamma = 0.2828$, $\eta = 0.5$ and $\tau_{i,j} = \lambda_{i,j}/16.6$.

In both of the deblurring and super-resolution experiments, 7 x 7 patches (for HR image) with 5-pixel-width overlap between adjacent patches were used in the proposed methods. For color images, all of the test methods were applied to the luminance component only because human visual system is more sensitive to luminance changes, and the bi-cubic interpolator was applied to the chromatic components. Here we only report the PSNR and SSIM [44] results for the luminance component. To examine more comprehensively the proposed approach, we give three results of the proposed method: the results by using only AS_DS (denoted by AS_DS), by using AS_DS plus AR regularization (denoted by AS_DS-AR), and by using AS_DS with both AR and nonlocal similarity regularization (denoted by AS_DS-AR-NL).2

C. Experimental Results on Deblurring

To verify the effectiveness of AS_DS and adaptive regularizations and the robustness of them to the training datasets, we first present the deblurring results on image Parrot by the proposed methods in Fig. 3. More PSNR and SSIM results can be found

2A website of this paper has been built, where all of the experimental results and the Matlab source code of the proposed algorithm can be downloaded. [Online]. Available: http://www4.comp.polyu.edu.hk/~cslzhang/AS_DS_AReg.htm
TABLE I

| Images          | [10]         | [42]         | [45]         | [46]         | [58]         | ASDS-TD1 | ASDS-TD2 | ASDS-AR-TD1 | ASDS-AR-NL-TD1 | ASDS-AR-NL-TD2 |
|-----------------|--------------|--------------|--------------|--------------|--------------|---------|---------|-------------|----------------|----------------|
| Barbara         | 25.83        | 25.59        | 26.11        | 26.28        | 27.90        | 26.60   | 26.65   | 26.93       | 26.99          | 27.63          | 27.70   |
|                 | 0.7492       | 0.7373       | 0.7580       | 0.7671       | 0.8171       | 0.7764  | 0.7709  | 0.7932      | 0.7893         | 0.8166         |         |
| Bike            | 23.09        | 24.24        | 24.38        | 24.15        | 24.77        | 25.29   | 25.50   | 25.21       | 25.40          | 25.32          | 25.48   |
|                 | 0.6959       | 0.7588       | 0.7564       | 0.7530       | 0.7740       | 0.8014  | 0.8082  | 0.7989      | 0.8052         | 0.8003         | 0.8069  |
| Straw           | 20.96        | 21.31        | 21.65        | 21.32        | 22.67        | 22.32   | 22.38   | 22.39       | 22.45          | 22.51          | 22.56   |
|                 | 0.4856       | 0.5415       | 0.5594       | 0.5322       | 0.6541       | 0.6594  | 0.6651  | 0.6563      | 0.6615         | 0.6549         | 0.6540  |
| Boats           | 28.80        | 28.94        | 29.44        | 29.81        | 29.90        | 28.85   | 28.94   | 29.40       | 29.48          | 30.73          | 30.76   |
|                 | 0.8274       | 0.8331       | 0.8459       | 0.8496       | 0.8528       | 0.8076  | 0.8039  | 0.8286      | 0.8272         | 0.8665         | 0.8670  |
| Parrots         | 27.80        | 28.80        | 28.96        | 29.04        | 30.22        | 30.71   | 30.90   | 30.64       | 30.79          | 30.76          | 30.92   |
|                 | 0.8652       | 0.8704       | 0.8722       | 0.8824       | 0.8906       | 0.8926  | 0.8926  | 0.8920      | 0.8933         | 0.8921         | 0.8939  |
| Baboon          | 21.06        | 21.16        | 21.33        | 21.21        | 21.46        | 21.43   | 21.45   | 21.56       | 21.55          | 21.62          | 21.62   |
|                 | 0.4811       | 0.5095       | 0.5192       | 0.5126       | 0.5315       | 0.5315  | 0.5386  | 0.5378      | 0.5385         | 0.5754         | 0.5765  |
| Hat             | 29.75        | 31.13        | 30.88        | 30.91        | 30.85        | 31.46   | 31.67   | 31.41       | 31.58          | 31.43          | 31.65   |
|                 | 0.8393       | 0.8624       | 0.8567       | 0.8591       | 0.8608       | 0.8700  | 0.8725  | 0.8692      | 0.8721         | 0.8689         | 0.8733  |
| Pentagon        | 24.69        | 25.12        | 25.57        | 25.26        | 26.00        | 25.58   | 25.62   | 25.88       | 25.89          | 26.41          | 26.46   |
|                 | 0.6452       | 0.6853       | 0.7020       | 0.6830       | 0.7210       | 0.7285  | 0.7290  | 0.7385      | 0.7380         | 0.7511         | 0.7539  |
| Camera-man      | 25.73        | 26.72        | 27.38        | 26.86        | 27.24        | 27.01   | 27.14   | 27.25       | 27.37          | 27.87          | 28.00   |
|                 | 0.8161       | 0.8330       | 0.8443       | 0.8361       | 0.8308       | 0.7956  | 0.7836  | 0.8255      | 0.8202         | 0.8578         | 0.8605  |
| Peppers         | 27.89        | 28.44        | 28.87        | 28.75        | 28.70        | 28.24   | 28.25   | 28.64       | 28.68          | 29.46          | 29.51   |
|                 | 0.8123       | 0.8131       | 0.8298       | 0.8274       | 0.8151       | 0.7748  | 0.7782  | 0.7992      | 0.7941         | 0.8357         | 0.8393  |
| Average         | 25.56        | 26.15        | 26.46        | 26.36        | 26.97        | 26.75   | 26.85   | 26.93       | 27.02          | 27.37          | 27.47   |
|                 | 0.7217       | 0.7443       | 0.7544       | 0.7500       | 0.7748       | 0.7695  | 0.7683  | 0.7789      | 0.7786         | 0.7910         | 0.7943  |

Fig. 4. Comparison of the deblurred images on Parrot by different methods (uniform blur kernel and $\sigma_n = \sqrt{2}$). Top row: original, degraded, method [10] ($\text{PSNR} = 27.80 \text{ dB}, \text{SSIM} = 0.8652$) and the method in [42] ($\text{PSNR} = 28.80 \text{ dB}, \text{SSIM} = 0.8704$). Bottom row: the method in [45] ($\text{PSNR} = 28.96 \text{ dB}, \text{SSIM} = 0.8722$), the method in [46] ($\text{PSNR} = 29.04 \text{ dB}, \text{SSIM} = 0.8824$), BM3D [58] ($\text{PSNR} = 30.22 \text{ dB}, \text{SSIM} = 0.8906$), and the proposed method ($\text{PSNR} = 30.92 \text{ dB}, \text{SSIM} = 0.8936$).

In Table I, from Fig. 3 and Table I, we can see that the proposed methods generate almost the same deblurring results with TD1 and TD2. We can also see that the ASDS method is effective in deblurring. By combining the adaptive regularization terms, the deblurring results can be further improved by eliminating the ringing artifacts around edges. Due to the page limit, we will only show the results by ASDS-AR-NL-TD2 in the following development.

The deblurring results by the competing methods are then compared in Figs. 4–6. One can see that there are many noise residuals and artifacts around edges in the deblurred images by the iterated wavelet shrinkage method [10]. The TV-based
methods in [42] and [45] are effective in suppressing the noises; however, they produce over-smoothed results and eliminate much image details. The $l_0$-norm sparsity-based method of [46] is very effective in reconstructing smooth image areas; however, it fails to reconstruct fine image edges. The BM3D method [58] is very competitive in recovering the image structures. However, it tends to generate some “ghost” artifacts around the edges (e.g., the image *Cameraman* in Fig. 6). The proposed method leads to the best visual quality. It can not only remove the blurring effects and noise, but also reconstruct more and sharper image edges than other methods. The excellent edge preservation owes to the adaptive sparse domain selection
strategy and adaptive regularizations. The PSNR and SSIM results by different methods are listed in Tables I–IV. For the experiments using uniform blur kernel, the average PSNR improvements of ASDS-AR-NL-TD2 over the second best method (i.e., BM3D [58]) are 0.50 dB (when $\sigma_n = \sqrt{2}$) and 0.40 dB (when $\sigma_n = 2$), respectively. For the experiments using Gaussian blur kernel, the PSNR gaps between all the competing methods become smaller, and the average PSNR improvements of ASDS-AR-NL-TD2 over the BM3D method are 0.15 dB (when $\sigma_n = \sqrt{2}$) and 0.18 dB (when $\sigma_n = 2$), respectively. We can also see that the proposed ASDS-AR-NL method achieves the highest SSIM index.

### D. Experimental Results on Single-Image Super-Resolution

Here, we present experimental results of single-image super-resolution. Again, we first test the robustness of the proposed method to the training dataset. Fig. 7 shows the reconstructed HR Parrot images by the proposed methods. We can see that

| Images | PSNR (dB) | SSIM | Previous Method | Proposed Method |
|--------|----------|------|----------------|-----------------|
| Barbara |          |      |                |                 |
| Bike   |          |      |                |                 |
| Straw  |          |      |                |                 |
| Boats  |          |      |                |                 |
| Parrots|          |      |                |                 |
| Baboon |          |      |                |                 |
| Hat    |          |      |                |                 |
| Peppers|          |      |                |                 |
| Average|          |      |                |                 |

---

**TABLE II**

| Images | PSNR (dB) | SSIM | Previous Method | Proposed Method |
|--------|----------|------|----------------|-----------------|
| Barbara |          |      |                |                 |
| Bike   |          |      |                |                 |
| Straw  |          |      |                |                 |
| Boats  |          |      |                |                 |
| Parrots|          |      |                |                 |
| Baboon |          |      |                |                 |
| Hat    |          |      |                |                 |
| Peppers|          |      |                |                 |
| Average|          |      |                |                 |

---

**TABLE III**

| Images | PSNR (dB) | SSIM | Previous Method | Proposed Method |
|--------|----------|------|----------------|-----------------|
| Barbara |          |      |                |                 |
| Bike   |          |      |                |                 |
| Straw  |          |      |                |                 |
| Boats  |          |      |                |                 |
| Parrots|          |      |                |                 |
| Baboon |          |      |                |                 |
| Hat    |          |      |                |                 |
| Peppers|          |      |                |                 |
| Average|          |      |                |                 |
TABLE IV
PSNR (dB) AND SSIM RESULTS OF DEBLURRED IMAGES (GAUSSIAN BLUR KERNEL, NOISE LEVEL $\sigma_n = 2$)

| Images   | [10] | [42] | [45] | [46] | [58] | ASDS-TD1 | ASDS-TD2 | ASDS-AR-TD1 | ASDS-AR-TD2 | ASDS-AR-NL-TD1 | ASDS-AR-NL-TD2 |
|----------|------|------|------|------|------|----------|----------|-------------|-------------|----------------|----------------|
| Barbara  | 23.57| 23.19| 23.07| 23.62| 23.70| 23.72    | 23.72    | 23.73       | 23.78       | 23.78          | 23.78          |
| Bike     | 21.58| 21.88| 20.97| 21.93| 22.53| 22.41    | 22.45    | 22.41       | 22.45       | 22.45          | 22.45          |
| Straw    | 20.10| 19.75| 19.24| 20.10| 20.81| 20.57    | 20.60    | 20.60       | 20.72       | 20.72          | 20.75          |
| Boats    | 25.87| 25.48| 24.63| 26.24| 26.71| 26.78    | 26.82    | 26.81       | 26.98       | 26.98          | 26.96          |
| Parrots  | 26.10| 25.92| 25.05| 26.38| 27.40| 27.08    | 27.14    | 27.13       | 27.24       | 27.24          | 27.50          |
| Baboon   | 20.16| 20.00| 19.79| 20.17| 20.28| 20.28    | 20.29    | 20.28       | 20.32       | 20.32          | 20.31          |
| Hat      | 27.94| 28.86| 28.27| 28.59| 28.67| 28.59    | 28.69    | 28.59       | 28.80       | 28.80          | 28.87          |
| Pentag-  |      |      |      |      |      |          |          |             |             |                |                |
| on       | 23.13| 22.46| 21.89| 23.13| 23.65| 23.69    | 23.69    | 23.69       | 23.80       | 23.80          | 23.81          |
| Camera   |      |      |      |      |      |          |          |             |             |                |                |
| Man      | 22.93| 23.23| 22.36| 23.25| 23.62| 23.72    | 23.76    | 23.76       | 23.95       | 23.95          | 23.95          |
| Peppers  |      |      |      |      |      |          |          |             |             |                |                |
|         | 25.72| 25.50| 24.38| 26.24| 26.44| 26.70    | 26.76    | 26.76       | 26.91       | 26.91          | 26.93          |
| Average  |      |      |      |      |      |          |          |             |             |                |                |

Fig. 7. Super-resolution results (scaling factor 3) on image Parrot by the proposed methods. Top row: original, LR image. ASDS-TD1 (PSNR = 29.47 dB, SSIM = 0.9031) and ASDS-TD2 (PSNR = 29.51 dB, SSIM = 0.9034). Bottom row: ASDS-AR-TD1 (PSNR = 29.61 dB, SSIM = 0.9036), ASDS-AR-TD2 (PSNR = 29.63 dB, SSIM = 0.9038), ASDS-AR-NL- TD1 (PSNR = 29.97 dB, SSIM = 0.9090), and ASDS-AR-NL-TD2 (PSNR = 30.00 dB, SSIM = 0.9093).

the proposed method with the two different training datasets produces almost the same HR images. It can also be observed that the ASDS scheme can well reconstruct the image, while there are still some ringing artifacts around the reconstructed edges. Such artifacts can be reduced by coupling ASDS with the AR model based regularization, and the image quality can be further improved by incorporating the nonlocal similarity regularization.

Next we compare the proposed methods with state-of-the-art methods in [10], [25], [43], [47]. The visual comparisons are shown in Figs. 8 and 9. We see that the reconstructed HR images by the method in [10] have many jaggy and ringing artifacts.
Fig. 8. Reconstructed HR images (scaling factor 3) of Girl by different methods. Top row: LR image, method [10] (PSNR = 32.93 dB, SSIM = 0.8102) and the method in [47] (PSNR = 31.21 dB, SSIM = 0.7878). Bottom row: the method in [43] (PSNR = 31.94 dB, SSIM = 0.7704), the method in [25] (PSNR = 32.51 dB, SSIM = 0.7912) and the proposed method (PSNR = 33.53 dB, SSIM = 0.8242).

Fig. 9. Reconstructed HR images (scaling factor 3) of Parrot by different methods. Top row: LR image, method [10] (PSNR = 28.78 dB, SSIM = 0.8845) and the method in [47] (PSNR = 27.59 dB, SSIM = 0.8836). Bottom row: the method in [43] (PSNR = 27.71 dB, SSIM = 0.8682), the method in [25] (PSNR = 27.98 dB, SSIM = 0.8663) and the proposed method (PSNR = 30.00 dB, SSIM = 0.9093).

The TV-regularization-based method [47] is effective in suppressing the ringing artifacts, but it generates piecewise constant block artifacts. The Softcuts method [43] produces very smooth edges and fine structures, making the reconstructed image look unnatural. By sparsely coding the LR image patches with the learned LR dictionary and recovering the HR image patches
Fig. 10. Reconstructed HR images (scaling factor 3) of noisy Girl by different methods. Top row: LR image, method [10] (PSNR = 30.37 dB, SSIM = 0.7094) and the method in [47] (PSNR = 29.77 dB, SSIM = 0.7258). Bottom row: the method in [43] (PSNR = 31.40 dB, SSIM = 0.7480), the method in [25] (PSNR = 30.70 dB, SSIM = 0.7088) and the proposed method (PSNR = 31.80 dB, SSIM = 0.7590).

Fig. 11. Reconstructed HR images (scaling factor 3) of noisy Parrot by different methods. Top row: LR image, method [10] (PSNR = 27.01 dB, SSIM = 0.7901) and the method in [47] (PSNR = 26.77 dB, SSIM = 0.8084). Bottom row: the method in [43] (PSNR = 27.42 dB, SSIM = 0.8458), the method in [25] (PSNR = 26.82 dB, SSIM = 0.7769) and the proposed method (PSNR = 28.72 dB, SSIM = 0.8668).

with the corresponding HR dictionary, the sparsity-based method in [25] is very competitive in terms of visual quality. However, it is difficult to learn a universal LR/HR dictionary pair that can represent various LR/HR structure pairs. It is observed that the reconstructed edges by [25] are relatively smooth and some fine image structures are not recovered. The proposed
method generates the best visual quality. The reconstructed edges are much sharper than all the other four competing methods, and more image fine structures are recovered. Often in practice the LR image will be noise corrupted, which makes the super-resolution more challenging. Therefore, it is necessary to test the robustness of the super-resolution methods.
Fig. 12. Some example images in the established 1000-image dataset.

Table VII

| Method        | Uniform blur kernel $\sigma_n = \sqrt{2}$ | Uniform blur kernel $\sigma_n = 2$ | Gaussian blur kernel $\sigma_n = \sqrt{2}$ | Gaussian blur kernel $\sigma_n = 2$ |
|---------------|-------------------------------------------|-----------------------------------|-------------------------------------------|-----------------------------------|
| [38]          | 28.51 (0.8139)                            | 27.96 (0.7966)                    | 26.09 (0.7297)                            | 25.91 (0.7209)                    |
| [46]          | 28.26 (0.8081)                            | 27.41 (0.7763)                    | 25.63 (0.7072)                            | 25.37 (0.6934)                    |

Fig. 13. PSNR gain distributions of deblurring experiments. (a) Uniform blur kernel with $\sigma_n = 2$. (b) Uniform blur kernel with $\sigma_n = 2$. (c) Gaussian blur kernel with $\sigma_n = \sqrt{2}$. (d) Gaussian blur kernel with $\sigma_n = 2$.

to noise. We added Gaussian white noise (with standard deviation of 5) to the LR images, and the reconstructed HR images are shown in Figs. 10 and 11. We see that the method in [10] is sensitive to noise and there are serious noise-caused artifacts around the edges. The TV-regularization-based method [47] also generates many noise-caused artifacts in the neighborhood of edges. The Softcuts method [43] results in over-smoothed HR images. Since the sparse representation-based method [25] is followed by a back-projection process to remove the blurring effect, it is sensitive to noise and the performance degrades much in the noisy case. In contrast, the proposed method shows good robustness to noise. Not only the noise is effectively
TABLE VIII
AVERAGE PSNR AND SSIM RESULTS OF THE RECONSTRUCTED HR IMAGES ON THE 1000-IMAGE DATASET

| Method            | Noise level $\sigma_n=0$     | Noise level $\sigma_n=5$            |
|-------------------|-------------------------------|-------------------------------------|
| ASDS-AR-NL-TD2    | 27.53 (0.7975)                | 26.56 (0.7444)                      |
| [25]              | 26.26 (0.7444)                | 25.34 (0.6711)                      |
| [47]              | 26.09 (0.7705)                | 25.31 (0.7156)                      |

Fig. 14. PSNR gain distributions of super-resolution experiments. (a) Noise level $\sigma_n = 0$. (b) Noise level $\sigma_n = 5$.

TABLE IX
AVERAGE PSNR AND SSIM RESULTS BY THE PROPOSED ASDS-AR-NL-TD2 METHOD WITH DIFFERENT NUMBERS OF CLASSES ON THE 1000-IMAGE DATASET

| Number of classes | Deblurring with uniform blur kernel and $\sigma_n=\sqrt{2}$ | Super-resolution with noise level $\sigma_n=0$ |
|-------------------|----------------------------------------------------------|-----------------------------------------------|
| 100               | 29.29 (0.8379)                                           | 27.51 (0.7971)                                |
| 200               | 29.36 (0.8397)                                           | 27.52 (0.7974)                                |
| 400               | 29.31 (0.8380)                                           | 27.53 (0.7975)                                |

suppressed, but also the image fine edges are well reconstructed. This is mainly because the noise can be more effectively removed and the edges can be better preserved in the adaptive sparse domain. From Tables V and VI, we see that the average PSNR gains of ASDS-AR-NL-TD2 over the second best methods [10] (for the noiseless case) and [43] (for the noisy case) are 1.13 and 0.77 dB, respectively. The average SSIM gains over the methods [10] and [43] are 0.0348 and 0.021 for the noiseless and noisy cases, respectively.

E. Experimental Results on a 1000-Image Dataset

To more comprehensively test the robustness of the proposed IR method, we performed extensive deblurring and super-resolution experiments on a large dataset that contains 1000 natural images of various contents. To establish this dataset, we randomly downloaded 822 high-quality natural images from the Flickr website\(^3\) and selected 178 high-quality natural images from the Berkeley Segmentation Database.\(^4\) A $256 \times 256$ subimage that is rich in edge and texture structures was cropped from each of these 1000 images to test our method. Fig. 12 shows some example images in this dataset.

For image deblurring, we compared the proposed method with the methods in [46] and [58], which perform the second and third best in our experiments in Section VI-D. The average PSNR and SSIM values of the deblurred images by the test methods are shown in Table VII. To better illustrate the advantages of the proposed method, we also drew the distributions of its PSNR gains over the two competing methods in Fig. 13. From Table VII and Fig. 13, we can see that the proposed method constantly outperforms the competing methods for the uniform blur kernel, and the average PSNR gain over the BM3D [58] is up to 0.85 dB (when $\sigma_n = \sqrt{2}$). Although the performance gaps between different methods become much smaller for the nontruncated Gaussian blur kernel, it can still be observed that the proposed method mostly outperforms BM3D [58] and [46], and the average PSNR gain over BM3D [58] is up to 0.19 dB (when $\sigma_n = 2$). For image super-resolution, we compared the proposed method with the two methods in [25] and [47]. The average PSNR and SSIM values by the test methods are listed in Table VIII, and the distributions of PSNR gain of our method over [25] and [47] are shown in Fig. 14. From Table VIII and Fig. 14, we can see that the proposed method performs constantly better than the competing methods.

With this large dataset, we tested the robustness of the proposed method to the number of classes in learning the subdictionaries and AR models. Specifically, we trained the subdictionaries and AR models with different numbers of classes, i.e., 100, 200, and 400, and applied them to the established 1000-image dataset. Table IX presents the average PSNR and SSIM values of the restored images. We can see that the three different numbers of classes lead to very similar image deblurring and super-resolution performance. This illustrates the robustness of the proposed method to the number of classes.

\(^3\)[Online]. Available: http://www.flickr.com/
\(^4\)[Online]. Available: http://www.eecs.berkeley.edu/Research/Projects/CS/vision/grouping/segbench.
Another important issue of the proposed method is the size of image patch. Clearly, the patch size cannot be big; otherwise, they will not be micro-structures and hence cannot be represented by a small number of atoms. To evaluate the effects of the patch size on IR results, we trained the subdictionaries and AR models with different patch sizes, i.e., 3 × 3, 5 × 5 and 7 × 7. Then we applied these subdictionaries and AR models to the ten test images and the constructed 1000-image database. The experimental results of deblurring and super-resolution are presented in Tables X–XII, from which we can see that these different patch sizes lead to similar PSNR and SSIM results. However, it can be found that the smaller patch sizes (i.e., 3 × 3 and 5 × 5) tend to generate some artifacts in smooth regions, as shown in Fig. 15. Therefore, we adopt 7 × 7 as the image patch size in our implementation.

F. Discussions on the Computational Cost

In Algorithm 1, the matrices \( \mathbf{U} \) and \( \mathbf{V} \) are sparse matrices, and can be precalculated after the initialization of the AR models and the nonlocal weights. Hence, Step 2a) can be executed fast. For image deblurring, the calculation of \( \mathbf{U}_{\alpha}^{(k)} \) can be implemented by FFT, which is faster than direct matrix calculation. Steps 2b) and 2d) require \( \mathcal{O}(n^2) \) multiplications, where \( n \) is the number of pixels of each patch and \( N \) is the number of patches. In our implementation, \( N \approx N_P/4 \), where \( N_P \) is the number of pixels of the entire image. Since each patch can be sparsely coded individually, Steps 2b) and 2d) can be executed in parallel to speed up the algorithm. The update of subdictionaries and AR models requires \( \mathcal{O} \) operations of nearest neighbor search. We update them in every \( P \) iterations (\( P = 100 \) in our implementation) to speed up Algorithm 1. As an iterative shrinkage

### Table X

| Patch Size | Barbara | Bike | Straw | Boats | Parrots | Baboon | Hat | Pentagon | Cameraman | Peppers | Average |
|------------|---------|------|-------|-------|---------|--------|-----|----------|-----------|---------|---------|
| 3×3        | 27.33   | 0.7936 | 25.68 | 22.32 | 30.64   | 31.07  | 21.61| 32.12    | 26.44     | 28.09  | 29.55  | 27.49  |
| 5×5        | 27.59   | 0.8116 | 25.54 | 22.44 | 30.81   | 31.04  | 21.61| 31.84    | 26.48     | 28.11  | 29.63  | 27.51  |
| 7×7        | 27.70   | 0.8192 | 25.48 | 22.56 | 30.76   | 30.92  | 21.62| 31.65    | 26.46     | 28.00  | 29.51  | 27.47  |

### Table XI

| Patch Size | Girl | Parrot | Butterfly | Leaves | Parthenon | Flower | Hat | Raccoon | Bike | Plants | Average |
|------------|------|--------|-----------|--------|-----------|--------|-----|---------|------|--------|---------|
| 3×3        | 33.55 | 0.8251 | 29.96    | 27.28 | 27.00    | 26.84  | 29.27| 30.95    | 29.18 | 24.46  | 33.54  | 29.20  |
| 5×5        | 33.56 | 0.8240 | 30.09    | 27.39 | 27.00    | 26.90  | 29.25| 31.10    | 29.22 | 24.53  | 33.59  | 29.26  |
| 7×7        | 33.55 | 0.8204 | 30.14    | 27.34 | 26.93    | 26.89  | 29.19| 31.04    | 29.24 | 24.62  | 33.37  | 29.22  |

### Table XII

| Patch Size | Deblurring with uniform blur kernel and \( \sigma_n = \sqrt{2} \) | Super-resolution with noise level \( \sigma_n = 0 \) |
|------------|-------------------------------------------------------------|-------------------------------------------------------|
| 3×3        | 29.60 (0.8466)                                              | 27.51 (0.7979)                                         |
| 5×5        | 29.56 (0.8450)                                              | 27.54 (0.7984)                                         |
| 7×7        | 29.36 (0.8397)                                              | 27.53 (0.7976)                                         |
algorithm, the proposed Algorithm 1 converges in 700 to 1000 iterations in most cases. For a 256 × 256 image, the proposed algorithm requires about 2 × 5 min for image deblurring and super-resolution on an Intel Core2 Duo 2.79G PC under the MATLAB R2010a programming environment. In addition, several accelerating techniques, such as those in [51] and [52], can be used to accelerate the convergence of the proposed algorithm. Hence, the computational cost of the proposed method can be further reduced.

VII. CONCLUSION

We proposed a novel sparse representation-based image deblurring and (single-image) super-resolution method using adaptive sparse domain selection (ASDS) and adaptive regularization (AReg). Considering the fact that the optimal sparse domains of natural images can vary significantly across different images and different image patches in a single image, we selected adaptively the dictionaries that were prelearned from a dataset of high-quality example patches for each local patch. The ASDS improves significantly the effectiveness of sparse modeling and consequently the results of image restoration. To further improve the quality of reconstructed images, we introduced two AReg terms into the ASDS based image restoration framework. A set of autoregressive (AR) models were learned from the training dataset and were used to regularize the image local smoothness. The image nonlocal similarity was incorporated as another regularization term to exploit the image nonlocal redundancies. An iterated shrinkage algorithm was proposed to implement the proposed ASDS algorithm with AReg. The experimental results on natural images showed that the proposed ASDS-AReg approach outperforms many state-of-the-art methods in both PSNR and visual quality.

REFERENCES

[1] M. Bertero and P. Boccacci, Introduction to Inverse Problems in Imaging. Bristol, U.K.: IOP, 1998.
[2] D. Donoho, “Compressed sensing,” IEEE Trans. Inf. Theory, vol. 52, no. 4, pp. 1289–1306, Apr. 2006.
[3] E. Candès and T. Tao, “Near optimal signal recovery from random projections: Universal encoding strategies?,” IEEE Trans. Inf. Theory, vol. 52, no. 12, pp. 5406–5425, Dec. 2006.
[4] E. Candès, J. Romberg, and T. Tao, “Robust uncertainty principles: Exact signal reconstruction from incompletely known frequency information,” IEEE Trans. Inf. Theory, vol. 52, no. 2, pp. 489–509, Feb. 2006.
[5] M. R. Banham and A. K. Katsaggelos, “Digital image restoration,” IEEE Trans. Signal Process. Mag., vol. 14, no. 2, pp. 24–41, Mar. 1997.
[6] L. Rudin, S. Osher, and E. Fatemi, “Nonlinear total variation based noise removal algorithms,” Phys. D, vol. 60, pp. 259–268, 1992.
[7] T. Chan, S. Esedoglu, F. Park, and A. Yip, “Recent developments in total variation image restoration,” in Mathematical Models of Computer Vision, N. Paragios, Y. Chen, and O. Faugeras, Eds. New York: Springer-Verlag, 2005.
[8] S. Kindermann, S. Osher, and P. W. Jones, “Deblurring and denoising of images by nonlocal functionals,” Multiscale Model. Simul., vol. 4, no. 4, pp. 1091–1115, 2005.
[9] R. Molina, J. Mateos, and A. K. Katsaggelos, “Blind deconvolution using a variational approach to parameter, image, and blur estimation,” IEEE Trans. Image Process., vol. 15, no. 12, pp. 3715–3727, Dec. 2006.
[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, pp. 1431–1457, 2004.
[11] P. Combettes and V. Wajs, “Signal recovery by proximal forward-backward splitting,” Siam J. Multiscale Model. Simul., vol. 4, pp. 1168–1200, 2005.
[12] J. M. Bioucas Dias and M. A. T. Figueiredo, “A new TwIST: Two-step iterative shrinkage/thresholding algorithms for image restoration,” IEEE Trans. Image Process., vol. 16, no. 12, pp. 2992–3004, Dec. 2007.
[13] M. Elad, M. A. T. Figueiredo, and Y. Ma, “On the role of sparse and redundant representations in image processing,” Proc. IEEE, vol. 98, no. 6, pp. 972–982, Jun. 2010.
[14] J. Mairal, M. Elad, and G. Sapiro, “Sparse representation for color image restoration,” IEEE Trans. Image Process., vol. 17, no. 1, pp. 53–69, Jan. 2008.
[15] J. Mairal, F. Bach, J. Ponce, G. Sapiro, and A. Zisserman, “Non-local sparse models for image restoration,” in Proc. Int. Conf. Comput. Vis., Tokyo, Japan, 2009, pp. 2272–2279.
[16] M. Irani and S. Peleg, “Edge detection by soft thresholding,” IEEE Trans. Inf. Theory, vol. 41, no. 3, pp. 613–627, May 1995.
[17] 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.
[18] M. I. Fadili and J. M. Starck, “Sparse representation-based image deconvolution by iterative thresholding,” in Proc. Astronom. Data Anal. IV, Marseille, France, Sep. 2006.
[19] J. Bobin, J. Starck, J. Fadili, Y. Moudden, and D. Donoho, “Morphological component analysis: An adaptive thresholding strategy,” IEEE Trans. Image Process., vol. 16, no. 11, pp. 2675–2681, Nov. 2007.
[20] J. Yang, J. Wright, Y. Ma, and T. Huang, “Image super-resolution as sparse representation of raw image patches,” in Proc. IEEE Comput. Vis. Pattern Recognit., Jun. 2008, pp. 1–8.
[21] 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.
[22] J. Mairal, G. Sapiro, and M. Elad, “Learning multiscale sparse representations for image and video restoration,” SIAM Multiscale Model. Simul., vol. 7, no. 1, pp. 214–241, Apr. 2008.
[23] R. Rubinstein, M. Zibulevsky, and M. Elad, “Double sparsity: Learning sparse dictionaries for sparse signal approximation,” IEEE Trans. Signal Process., vol. 58, no. 3, pp. 1553–1564, Mar. 2010.
[24] J. Mairal, F. Bach, J. Ponce, G. Sapiro, and A. Zisserman, “Supervised dictionary learning,” in Proc. Adv. Neural Inf. Process. Syst. Conf., 2008, pp. 1033–1040.
[25] G. Monaci and P. Vanderheyden, “Learning structured dictionaries for image representation,” in Proc. IEEE Int. Conf. Image Process., 2004, pp. 2351–2354.
[26] R. Rubinstein, A. M. Bruckstein, and M. Elad, “Dictionaries for sparse representation modeling,” IEEE Proc. IEEE, vol. 98, no. 6, pp. 1045–1057, Jun. 2010.
[27] B. K. Gunturk, A. U. Batur, Y. Altunbasak, M. H. Hayes, III, and R. M. Mersereau, “Eigenface-based super-resolution for face recognition,” in Proc. Int. Conf. Image Process., Oct. 2002, pp. 845–848.
[28] X. Wu, K. U. Barthel, and W. Zhang, “Piecewise 2-D autoregression for predictive image coding,” in Proc. Int. Conf. Image Process., Oct. 1998, vol. 3, pp. 901–904.
[29] X. Li and M. T. Orchard, “New edge-directed interpolation,” IEEE Trans. Image Process., vol. 10, no. 10, pp. 1521–1527, Oct. 2001.
[30] X. Zhang and X. Wu, “Image interpolation by 2-D autoregressive modeling and soft-decision estimation,” IEEE Trans. Image Process., vol. 17, no. 6, pp. 887–896, Jun. 2008.
[31] A. Buades, B. Coll, and J. M. Morel, “A review of image denoising algorithms, with a new one,” Multiscale Model. Simul., vol. 4, no. 2, pp. 490–530, 2005.
[32] K. Fukunaga, Introduction to Statistical Pattern Recognition, 2nd ed. New York: Academic, 1991.
[33] L. Zhang, R. Lukac, X. Wu, and D. Zhang, “PCA-based spatially adaptive denoising of CFA images for single-lens digital cameras,” IEEE Trans. Image Process., vol. 18, no. 4, pp. 797–812, Apr. 2009.
[34] L. Zhang, W. Dong, D. Zhang, and G. Shi, “Two-stage image denoising by principal component analysis with local pixel grouping,” Pattern Recognit., vol. 43, pp. 1531–1549, Apr. 2010.
[35] X. Wu, X. Zhang, and J. Wang, “Model-guided adaptive recovery of compressive sensing,” in Proc. Data Compression Conf., 2009, pp. 123–132.
DONG et al.: IMAGE DEBLURRING AND SUPER-RESOLUTION BY ASDS AND ADAPTIVE REGULARIZATION 1857

Dong received the B.S. degree in electronic engineering from the Huazhong University of Science and Technology, Wuhan, China, in 2004, and the Ph.D. degree in circuits and system from the Xidian University, Xi’an, China, in 2010.

From September to December 2006, he was a Visiting Student with Microsoft Research Asia, Beijing, China. From January 2009 to June 2010, he was a Research Assistant with the Department of Computer Science, The Hong Kong Polytechnic University, Hong Kong. Since September 2010, he has been a Lecture with The Hong Kong Polytechnic University, Hong Kong.

His research interests include image and video processing, computer vision, pattern recognition, multisensor data fusion, and optimal estimation theory. He is the lead guest editor of EURASIP Journal on Image and Video Processing, Special Issue on Emerging Methods for Color Image and Video Quality Enhancement, and the guest editor of Sensors, Special Issue on Hand-Based Biometrics Sensors and System.

Dr. Dong was the recipient of the Best Paper Award at SPIE VCIP2010.

Lei Zhang (M’04) received the B.S. degree from Shenyang Institute of Aeronautical Engineering, Shenyang, China, in 1995, and the M.S. and Ph.D. degrees in automatic control theory and engineering from Northwestern Polytechnical University, Xi’an, China, in 1998 and 2001, respectively.

From 2001 to 2002, he was a Research Associate with the Department of Computing, The Hong Kong Polytechnic University, Hong Kong. From January 2003 to January 2006, he was a Postdoctoral Fellow with the Department of Electrical and Computer Engineering, McMaster University, Hamilton, ON, Canada. In 2006, he joined the Department of Computing, The Hong Kong Polytechnic University, as an Assistant Professor, where, since September 2010, he has been an Associate Professor. His research interests include image and video processing, biometrics, computer vision, pattern recognition, multisensor data fusion, and optimal estimation. He is the lead guest editor of EURASIP Journal on Image and Video Processing, Special Issue on Emerging Methods for Color Image and Video Quality Enhancement, and the guest editor of Sensors, Special Issue on Hand-Based Biometrics Sensors and System.

Dr. Zhang is an associate editor of the IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS—C. He was the recipient of the Merit Award, the Faculty Award in Research and Scholarly Activities 2010, and the Best Paper Award of SPIE VCIP2010.

Guangming Shi (SM’10) received the B.S. degree in automatic control, M.S. degree in computer control, and Ph.D. degree in electronic information technology from Xidian University, Xi’an, China, in 1985, 1988, and 2002, respectively.

He joined the School of Electronic Engineering, Xidian University, Xi’an, China, in 1988. From 1994 to 1996, as a Research Assistant, he cooperated with the Department of Electronic Engineering, University of Hong Kong. Since 2003, he has been a Professor with the School of Electronic Engineering, Xidian University, and in 2004 the head of National Instruction Base of Electrician & Electronic (NBEE). From June 2004 to 2006, he studied with the Department of Electronic Engineering, University of Illinois at Urbana-Champaign (UIUC). Presently, he is the Deputy Director of the School of Electronic Engineering, Xidian University, and the academic leader in the subject of circuits and systems. His research interests include compressed sensing, theory and design of multirate filter banks, image denoising, low-bit-rate image/video coding and implementation of algorithms for intelligent signal processing (using digital signal processing and field-programmable gate arrays). He has authored or coauthored over 60 research papers.

Xiaolin Wu (SM’96–F’10) received the B.Sc. degree from Wuhu University, Wuhu, China, in 1982, and the Ph.D. degree from the University of Calgary, Calgary, ON, Canada, in 1988, both in computer science.

He started his academic career in 1988 and has since been on the faculty at the University of Western Ontario, New York Polytechnic University, Brooklyn, and currently McMaster University, Hamilton, ON, where he is a Professor with the Department of Electrical and Computer Engineering, and where he holds the NSERC-DALSA Industrial Research Chair in Digital Cinema. His research interests include image processing, multimedia compression, joint source-channel coding, multiple description coding, and network-aware visual communication. He has authored or coauthored over 180 research papers and holds two patents in these fields.

Prof. Wu currently serves as an associate editor for the IEEE TRANSACTIONS ON IMAGE PROCESSING.