

Sparse Representation based Image Interpolation with Nonlocal Autoregressive Modeling

Weisheng Dong^a, Lei Zhang^{b,1}, *Member, IEEE*
Rastislav Lukac^c, *Senior Member, IEEE*, and Guangming Shi^a, *Member, IEEE*

^aKey Laboratory of Intelligent Perception and Image Understanding (Chinese Ministry of Education)
School of Electronic Engineering, Xidian University, China

^bDept. of Computing, The Hong Kong Polytechnic University, Hong Kong, China

^cFoveon, Inc. / Sigma Corp., San Jose, USA

Abstract: Sparse representation has proven to be a promising approach to image super-resolution, where the low resolution (LR) image is usually modeled as the down-sampled version of its high resolution (HR) counterpart after blurring. When the blurring kernel is the Dirac delta function, i.e., the LR image is directly down-sampled from its HR counterpart without blurring, the super-resolution problem becomes an image interpolation problem. In such case, however, the conventional sparse representation models (SRM) become less effective because the data fidelity term will fail to constrain the image local structures. In natural images, fortunately, the many nonlocal similar patches to a given patch could provide nonlocal constraint to the local structure. In this paper we incorporate the image nonlocal self-similarity into SRM for image interpolation. More specifically, a nonlocal autoregressive model (NARM) is proposed and taken as the data fidelity term in SRM. We show that the NARM induced sampling matrix is less coherent with the representation dictionary, and consequently makes SRM more effective for image interpolation. Our extensive experimental results demonstrated that the proposed NARM based image interpolation method can effectively reconstruct the edge structures and suppress the jaggy/ringing artifacts, achieving the best image interpolation results so far in term of PSNR as well as perceptual quality metrics such as SSIM and FSIM.

Index Terms: Image interpolation, super-resolution, sparse representation, nonlocal autoregressive model.

¹ Corresponding author. Email: cszhang@comp.polyu.edu.hk.

1. Introduction

Image super-resolution has wide applications in digital photography, medical imaging, computer vision, and consumer electronics, aiming at reconstructing a high resolution (HR) image from its low resolution (LR) counterpart. As a typical inverse problem, image super-resolution can be modeled as $\mathbf{y}=\mathbf{DHx}+\mathbf{v}$, where \mathbf{x} is the unknown original image, \mathbf{H} is the blurring operator, \mathbf{D} is the down-sampling operator, \mathbf{v} is the additive noise, and \mathbf{y} is the observed data. In this paper we will focus on the situation that the observation is noise free and the blur kernel is the Dirac delta function, i.e., \mathbf{v} is zero and \mathbf{H} is the identity matrix. In this case, we have $\mathbf{y}=\mathbf{Dx}$; that is, \mathbf{y} is directly down-sampled from the original image \mathbf{x} , and the super-resolution problem becomes an image interpolation problem. A variety of image interpolation algorithms have been developed, including the classical bilinear and bi-cubic interpolators [1-2], the edge guided interpolators [3-5], the recently developed sparse coding based methods [6-7] and the sparse mixing estimators [8].

Reconstructing \mathbf{x} from its linear measurement $\mathbf{y}=\mathbf{DHx}+\mathbf{v}$ is an ill-posed inverse problem. The classical iterative back-projection (IBP) algorithm [9] reconstructs \mathbf{x} by minimizing $\hat{\mathbf{x}} = \arg \min_{\mathbf{x}} \|\mathbf{y} - \mathbf{DHx}\|_2^2$. However, the solution to this l_2 -minimization problem is generally not unique, and the reconstructed image by IBP is often noisy. To refine the solution space, some regularization terms of \mathbf{x} , denoted by $R(\mathbf{x})$, can be introduced to regularize the solution: $\hat{\mathbf{x}} = \arg \min_{\mathbf{x}} \{\|\mathbf{y} - \mathbf{DHx}\|_2^2 + \lambda \cdot R(\mathbf{x})\}$, where λ is a scalar constant. One widely used regularizer is the total variation (TV) model [10-13], which assumes that natural images have small first derivatives. However, the TV model favors piecewise constant image structures, and hence tends to smooth much the image details.

In recent years, the sparse representation models (SRM) [14-29] have shown promising results in image super-resolution. SRM assumes that the image \mathbf{x} is sparse in some domain spanned by a dictionary Ψ , i.e., $\mathbf{x} \approx \Psi\boldsymbol{\alpha}$ and most of the coefficients in $\boldsymbol{\alpha}$ are close to zero. Intuitively, the SRM regularizer can be set as $R(\mathbf{x}) = \|\boldsymbol{\alpha}\|_0$. However, the l_0 -minimization is non-convex. As the closest convex relaxation [28], the l_1 -norm regularizer $R(\mathbf{x}) = \|\boldsymbol{\alpha}\|_1$ is widely adopted, leading to the following SRM based super-resolution model:

$$\hat{\boldsymbol{\alpha}} = \arg \min_{\boldsymbol{\alpha}} \left\{ \|\mathbf{y} - \mathbf{DH\Psi\boldsymbol{\alpha}}\|_2^2 + \lambda \cdot \|\boldsymbol{\alpha}\|_1 \right\}. \quad (1)$$

Once the coding vector $\hat{\alpha}$ is obtained, the desired HR image can be reconstructed as $\hat{x} = \Psi\hat{\alpha}$. The above l_1 -minimization problem can be solved by techniques such as the iterative shrinkage based surrogate [16] and proximal [17] algorithms. In addition to the standard sparse coding model in Eq. (1), in [44, 33, 26] group/simultaneous sparse coding models were proposed to exploit the nonlocal self-similarity and local sparsity in natural images. In [27], a centralized sparse coding model was also proposed to exploit the image nonlocal redundancy.

The SRM based super-resolution has a close relationship with the compressive sensing (CS) theory [24, 25, 45]. According to the CS theory, in order to accurately recover the original signal x from its linear measurement y , the following two conditions should be satisfied.

1) *Incoherence*: the linear observation matrix $A=DH$ and the dictionary Ψ should be incoherent. The *coherence* between A and Ψ can be computed by $\mu(A, \Psi) = \sqrt{n} \cdot \max_{1 \leq k, j \leq n} |\langle a_k, \psi_j \rangle|$ [45], where n is the sample size, a_k denotes the k^{th} row of A and ψ_j denotes the j^{th} atom (i.e., column) of Ψ . There is $\mu(A, \Psi) \in [1, \sqrt{n}]$. In order for a good reconstruction, the coherence μ should be small.

2) *Sparsity*: the original signal x should have a sparse expansion over the dictionary Ψ .

In the case of image interpolation, the matrix H is the identity matrix, and hence the sampling matrix $A=D$ is the canonical or spike matrix. It has been shown [45] that the coherence between such a D and Ψ is minimal, i.e., $\mu(D, \Psi) = 1$, when Ψ is the Fourier dictionary. However, natural images are generally not band-limited due to the many sharp edge structures, and thus the Fourier dictionary Ψ may not lead to sparse enough representations of natural images. For other dictionaries such as the wavelet dictionary, the canonical sampling matrix D is generally coherent with Ψ . For instances, the average coherence values between canonical sampling matrix D and the Haar wavelet dictionary, the Daubechies D4 and D8 dictionaries are 5.65, 5.72 and 5.76, respectively, when the sample size is $n=64$ (i.e., 8×8 image patch). Considering that the maximal coherence is $\sqrt{64}=8$ for $n=64$, we can see that the wavelet dictionary is highly coherent with the canonical sampling matrix. Apart from the analytical Fourier and wavelet dictionaries, by using algorithms such as K-SVD [30] we can learn redundant dictionaries $\Psi \in \mathfrak{R}^{64 \times 256}$ from some high quality natural images. The coherence between the canonical sampling matrix D and such learned

dictionaries are about 4.45~5.97. Overall, we can see that the coherence between the canonical sampling matrix D and the dictionary Ψ is high, making the SRM based image interpolation less effective.

Some recently developed SRM based image interpolation/super-resolution methods [6, 8, 50] use different models from Eq. (1) to reconstruct the HR image. In [6], a pair of over-complete dictionaries, denoted by Ψ_h and Ψ_l , are co-learned from a set of HR image patches and the associated set of LR image patches. It is enforced that the two sets of training patches share the same sparse representation coefficients over the couple of dictionaries. Then for an input LR image y , it is sparsely represented over Ψ_l , and the obtained sparse representation coefficients α are used to reconstruct the HR image x via $x = \Psi_h \alpha$. An improved version of this coupled dictionary learning based image super-resolution method was later presented in [50]. In [8], a series of linear inverse estimators of x are computed based on different priors on the image regularity. These estimators are then mixed in a frame over patches of coefficients with appropriate regularity, providing an l_1 -sparse representation weighted by the signal regularity in each patch. The sparse mixing of these inverse estimators improves the stability of each estimator and makes the final estimation more accurate.

In this paper, we propose a novel SRM based image interpolation approach by modeling and exploiting adaptively the image local and nonlocal redundancies². Natural images often show a rich amount of repetitive patterns. For a given image patch, we may find many similar patches to it, which can be spatially either close to or far from this patch. Such kind of nonlocal similarity is very helpful to improve the image restoration output, and it has been successfully used in image denoising [31-34], deblurring [35-36] and super-resolution [37-38, 49]. Considering the fact that a given pixel can be well approximated by its nonlocal neighbors, which is the underlying principle of nonlocal means filtering [31-34], we propose the concept of nonlocal autoregressive model (NARM), which refers to modeling a given pixel as the linear combination of its nonlocal neighboring pixels. The NARM can be viewed as a natural extension of the commonly used autoregressive model, which approximates a pixel as the linear combination of its local neighbors. The NARM reflects the image self-similarity, and it constrains the image local structure (i.e., the local patch) by using the nonlocal redundancy. On the other hand, the NARM can act as a kernel, and can be

² Since the local redundancy can also be viewed as nonlocal redundancy (just with shorter spatial distance), in the remaining of this paper we simply use nonlocal redundancy to represent both of them.

embedded into the data fidelity term of the conventional SRM model. Our study shows that the embedding of NARM kernels makes the sampling matrix more incoherent with the dictionary Ψ , which consequently enhances the effectiveness of SRM in image reconstruction according to the CS theory [24, 25, 45]. Nonetheless, the NARM induced kernels have very different physical meanings from the conventional kernels (such as Gaussian kernel) induced by the imaging system. The former mainly depends on the image content, while the later is mainly determined by the imaging process (e.g., lens, motion, etc).

By introducing and embedding NARM into SRM, the image interpolation problem can be generalized to the conventional SRM based image restoration problem. In addition to the sparsity prior of representation coefficients, we also assume that the nonlocal similar patches have similar coding coefficients. This can improve much the stability and accuracy of sparse coding. The variable splitting optimization technique [46] is adopted to effectively solve the proposed NARM based SRM model. Our experimental results on benchmark test images clearly demonstrate that the proposed NARM method outperforms much the classical bi-cubic interpolator [1-2], the representative edge-guided interpolators [3-5], and the recently developed SRM based image interpolation methods [6, 8] in term of PSNR, SSIM [42] and FSIM [43] measures, as well as visual perception quality.

The rest of the paper is organized as follows. Section 2 describes the NARM modeling for image interpolation. Section 3 discusses the NARM based SRM. Section 4 presents in detail the algorithm. Section 5 presents extensive experimental results and Section 6 concludes the paper.

2. Nonlocal Autoregressive Modeling

For image interpolation, it is assumed that the low-resolution (LR) image is directly down-sampled from the original high-resolution (HR) image. Let's denote by $y(n,m)$, $n=0,1,\dots,N_I-1$, $m=0,1,\dots,M_I-1$, the LR pixels and by $x(r,c)$, $r=0,1,\dots,s\cdot N_I-1$, $j=0,1,\dots,s\cdot M_I-1$, the HR pixels, where s is the scaling factor. We have $x(r,c) = y(n,m)$ when $r=s\cdot n$ and $c=s\cdot m$. Fig. 1 illustrates the case when $s=2$. The black dots represent the available LR pixels, while the white dots represent the missing pixels in the HR image. We can see that there is a great degree of freedom in recovering the missing pixels. One great challenge of image interpolation is to preserve the sharpness of edges in the interpolated HR image. The traditional edge-based interpolators preserve the

edge sharpness by interpolating the missing pixels along edges [3-5] instead of across the edges. However, it is rather challenging to accurately estimate the edge directions or local statistics (e.g., local covariance matrix) from the LR image, and thus artifacts may be generated in the interpolated HR image.

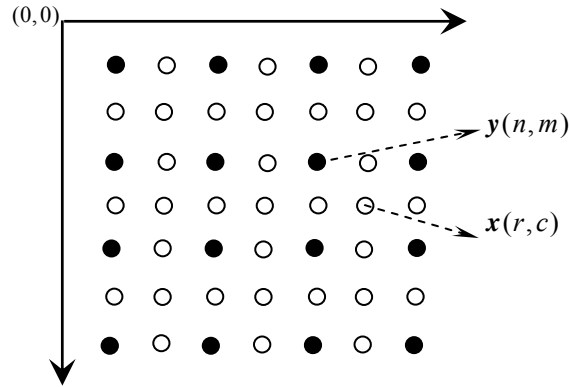


Figure 1: The low resolution (LR) image is down-sampled from its high resolution (HR) counterpart. The black dots represent the pixels of the LR image and the white dots represent the missing HR samples.



Figure 2: SRM based image interpolation (scaling factor: 3). (a) Original image; reconstructed images by (b) standard SRM with DCT dictionary (PSNR=32.49 dB); (c) standard SRM with local PCA dictionary (PSNR=32.50 dB); and (d) NARM based SRM with local PCA dictionary (PSNR=33.40 dB).

In this paper we aim to develop an SRM based image interpolation method. As we discussed in the Introduction Section, the canonical sampling matrix \mathbf{D} in image interpolation is generally coherent with the dictionary Ψ , making the standard SRM (refer to Eq. (1)) less effective for image interpolation. Fig. 2 shows an example. One can see that the standard SRM with either DCT dictionary or local PCA dictionary (refer to Sec. III for details) results in serious artifacts of ringings and zippers.

To improve the SRM based image interpolation, we propose to improve the observation model $\mathbf{y}=\mathbf{D}\mathbf{x}$ by incorporating the nonlocal self-similarity constraint. Since natural images have high local redundancy, many interpolation methods, including the classical bi-linear and bi-cubic interpolators [1-2] and the edge guided interpolators [3-5], interpolate the missing HR pixel, denoted by x_i , as the weighted average of its local

neighbors. In [5] the autoregressive model (ARM) is used to exploit the image local correlation for interpolation. Nonetheless, the image local redundancy may not be high enough for a faithful image reconstruction, especially in the region of edges, and thus artifacts such as ringings, jags and zippers often appear in the interpolated image. Apart from local redundancy, fortunately, natural images also have a rich amount of nonlocal redundancy (e.g., repetitive image structures across the image). The pixel x_i may have many nonlocal neighbors which are similar to it but spatially far from it. Clearly, those nonlocal neighbors (including the local ones) of x_i , denoted by x_i^j , can be used to approximate x_i by weighted average:

$$x_i \approx \sum_j \omega_i^j x_i^j. \quad (2)$$

In recent years the nonlocal methods have been successfully used in many image restoration applications [31-38].

In practice, we use a local patch centered at x_i , denoted by \mathbf{x}_i , to identify the nonlocal neighbors of x_i by patch matching. Since the missing HR pixel x_i and some of its local neighbors are not available, we initially interpolate the HR image \mathbf{x} using methods such as the bi-cubic interpolator. Then we could search for the nonlocal similar patches to \mathbf{x}_i in a large enough window around \mathbf{x}_i . A patch \mathbf{x}_i^j is chosen as the similar patch to \mathbf{x}_i if $d_i^j = \|\mathbf{x}_i - \mathbf{x}_i^j\|_2^2 \leq t$, where t is a preset threshold, or \mathbf{x}_i^j is chosen if it is within the first J ($J=25$ in our experiments) most similar patches to \mathbf{x}_i . We can then determine the weights ω_i^j by solving the following regularized least-square problem:

$$\hat{\boldsymbol{\omega}}_i = \arg \min_{\boldsymbol{\omega}} \|\mathbf{x}_i - \mathbf{X} \boldsymbol{\omega}_i\|_2^2 + \gamma \|\boldsymbol{\omega}_i\|_2^2, \quad (3)$$

where $\mathbf{X} = [\mathbf{x}_i^1, \mathbf{x}_i^2, \dots, \mathbf{x}_i^J]$, $\boldsymbol{\omega}_i = [\omega_i^1, \omega_i^2, \dots, \omega_i^J]^T$ and γ is the regularization parameter. The use of regularization in Eq. (3) is to enhance the stability of the least square solution, because both the patch \mathbf{x}_i and its neighbors in \mathbf{X} are noisy due to the interpolation error. The solution of Eq. (3) can be readily obtained as

$$\hat{\boldsymbol{\omega}}_i = (\mathbf{X}^T \mathbf{X} + \gamma \mathbf{I})^{-1} \mathbf{X}^T \mathbf{x}_i. \quad (4)$$

In practice, we use the conjugate gradient method to efficiently solve Eq. (4).

With $\boldsymbol{\omega}_i$, we propose the nonlocal autoregressive model (NARM) of image \mathbf{x} as

$$\mathbf{x} = \mathbf{S} \mathbf{x} + \mathbf{e}_x, \quad (5)$$

where \mathbf{e}_x is the modeling error, and the NARM matrix \mathbf{S} is

$$\mathbf{S}(i,j) = \begin{cases} \omega_i^j, & \text{if } x_i^j \text{ is a nonlocal neighbor of } x_i \\ 0, & \text{otherwise} \end{cases} \quad (6)$$

It can be seen that NARM is a natural extension and generalization of the traditional ARM that uses only the spatially local neighbors to approximate x_i . Fig. 3 illustrates the proposed NARM. We can see that a patch x_0 centered at x_0 has a bunch of nonlocal neighbors x_0^1, \dots, x_0^5 , and the weights assigned to the five neighbors are $\omega_0^1, \dots, \omega_0^5$. Meanwhile, anyone of the neighbors, for example x_0^1 , has its own nonlocal neighbors and the associated weights. Like that with traditional ARM, with NARM all the pixels in the image can be connected but with a much more complex graph.

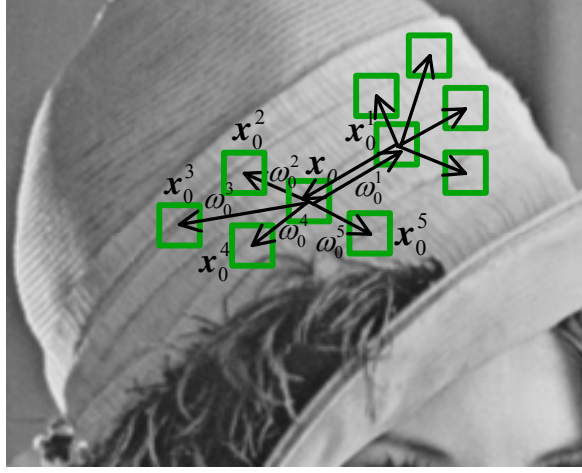


Figure 3: Illustration of the nonlocal autoregressive modeling (NARM).

The NARM in Eq. (5) can be embedded into the standard SRM (Eq. (1)), leading to a new data fidelity term and making the SRM effective for image interpolation. Applying the downsampling operator \mathbf{D} to Eq. (5), we have $\mathbf{y}=\mathbf{D}\mathbf{S}\mathbf{x}+\mathbf{e}_y$, where $\mathbf{y}=\mathbf{D}\mathbf{x}$ and $\mathbf{e}_y=\mathbf{D}\mathbf{e}_x$. The NARM based SRM for image interpolation can be generally written as

$$\hat{\boldsymbol{\alpha}} = \arg \min_{\boldsymbol{\alpha}} \left\{ \|\mathbf{y} - \mathbf{D}\mathbf{S}\boldsymbol{\Psi}\boldsymbol{\alpha}\|_2^2 + \lambda \cdot R(\boldsymbol{\alpha}) \right\} \text{ s.t. } \mathbf{y}=\mathbf{D}\boldsymbol{\Psi}\boldsymbol{\alpha}, \quad (7)$$

where $R(\boldsymbol{\alpha})$ is the regularization term (e.g., sparsity constraint on $\boldsymbol{\alpha}$) and λ is the parameter balancing the data fidelity term and the regularization term. We can see that the NARM matrix \mathbf{S} is functionally similar to the blurring kernel matrix \mathbf{H} in conventional SRM, reducing the degree of freedom of unknown pixels. Nonetheless, the NARM induced kernel \mathbf{S} has very different physical meanings from the conventional kernel

\mathbf{H} which is often induced by the imaging system. The shape of \mathbf{S} mainly depends on the image content, while \mathbf{H} is basically determined by the imaging process (e.g., lens, motion, etc).

With NARM, the sampling matrix of SRM in Eq. (7) becomes $\mathbf{A}=\mathbf{D}\mathbf{S}$. Let's then compute the coherence value between \mathbf{A} and different dictionaries Ψ . For 8×8 patches (i.e., $n=64$), the coherence between \mathbf{A} and the Haar wavelet basis is about 1.24~4.72 (different \mathbf{S} will lead to different coherence values), and the coherence values between \mathbf{A} and D4 and D8 wavelet dictionaries are 1.21~4.77 and 1.25~4.74, respectively. For the local PCA dictionaries (refer to Sec. III for details about PCA dictionary), the coherence values between \mathbf{A} and them are about 1.05~4.08. We can see that the NARM matrix \mathbf{S} improves much the incoherence between sampling matrix and dictionaries, including the fixed wavelet dictionary and the adaptive local PCA dictionary. In Fig. 4 we show some PCA dictionaries, and compare the coherence values for the canonical sampling matrix \mathbf{D} and the NARM improved sampling matrix \mathbf{A} . In Fig. 2(d), we show the reconstructed HR image by NARM based SRM in Eq. (7) (we set $R(\alpha)=\|\alpha\|_1$) with local PCA dictionaries. It can be clearly seen that the image edges reconstructed by NARM based SRM are much sharper than those by standard SRM methods.

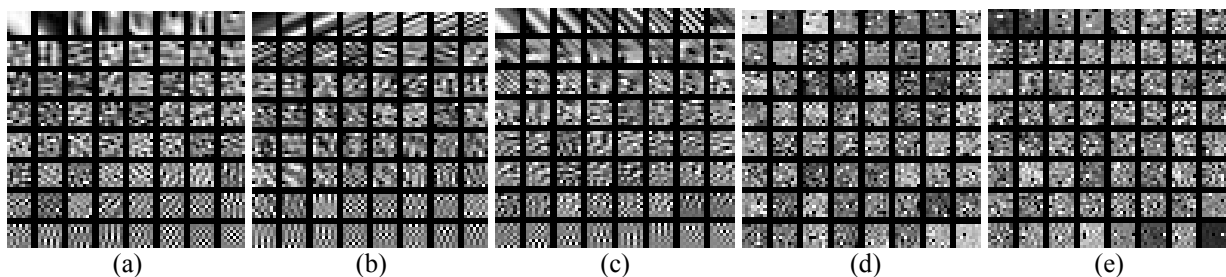


Figure 4: Examples of local PCA dictionaries (each block shows an atom of the local PCA dictionary) and their coherence values with different sampling matrices. Denote by μ_1 the coherence between the canonical sampling matrix \mathbf{D} and the local PCA dictionary, and by μ_2 the coherence between the NARM improved sampling matrix \mathbf{A} and the local PCA dictionary. (a) $\mu_1=3.78$, $\mu_2=1.60$; (b) $\mu_1=3.28$, $\mu_2=1.80$; (c) $\mu_1=3.22$, $\mu_2=2.08$; (d) $\mu_1=6.41$, $\mu_2=1.47$; (e) $\mu_1=7.50$, $\mu_2=1.69$. One can see that the NARM modeling improves the incoherence of the sparse representation model, and hence can lead to better image reconstruction results.

The calculation of \mathbf{S} needs to have the full resolution image \mathbf{x} , which is unknown and is to be reconstructed. In general, we start from some initial estimate of \mathbf{x} , denoted by $\mathbf{x}^{(0)}$, and then some estimation of \mathbf{S} , denoted by $\mathbf{S}^{(0)}$, can be obtained. By solving the NARM based SRM in Eq. (7), an updated version of \mathbf{x} , denoted by $\mathbf{x}^{(1)}=\Psi\hat{\alpha}$ can be obtained, from which the updated estimation of \mathbf{S} , denoted by $\mathbf{S}^{(1)}$, can be computed. Such a procedure is iterated until a desired estimation of \mathbf{x} is obtained.

3. NARM based Sparse Representation for Image Interpolation

To implement the NARM based SRM in Eq. (7), we need to set the regularization term $R(\boldsymbol{\alpha})$ and select the dictionary $\boldsymbol{\Psi}$. In this section, rather than using the conventional l_1 -regularization on $\boldsymbol{\alpha}$, a more effective regularization is used by considering the nonlocal correlation between sparse coding coefficients. Moreover, local PCA dictionaries are learned to better adapt to the local image structures.

A. The regularization term

Since natural images usually have sparse expansion over a dictionary of bases (e.g., DCT bases, wavelet bases, or some learned dictionary), the l_1 -norm sparsity regularization can well regularize the solution of Eq. (7). In patch based sparse representation, the image \mathbf{x} is partitioned into many (overlapped) patches \mathbf{x}_i , $i = 1, 2, \dots, N$, and each patch is coded over the dictionary $\boldsymbol{\Psi}$. Therefore, the NARM based SRM with l_1 -sparsity constraint can be written as

$$\hat{\boldsymbol{\alpha}} = \arg \min_{\boldsymbol{\alpha}} \left\{ \|\mathbf{y} - D\boldsymbol{S}\boldsymbol{\Psi}\boldsymbol{\alpha}\|_2^2 + \lambda \sum_{i=1}^N \|\boldsymbol{\alpha}_i\|_1 \right\} \text{ s.t. } \mathbf{y} = D\boldsymbol{\Psi}\boldsymbol{\alpha}, \quad (8)$$

where $\boldsymbol{\alpha}_i$ is the coding vector of patch \mathbf{x}_i , and $\boldsymbol{\alpha}$ is concatenated by all $\boldsymbol{\alpha}_i$. For convenience of expression, here we use $\mathbf{x} = \boldsymbol{\Psi}\boldsymbol{\alpha}$ to denote the representation of full image \mathbf{x} by combining all local patches $\mathbf{x}_i = \boldsymbol{\Psi}\boldsymbol{\alpha}_i$. Clearly, the correlation between sparse coding vectors $\boldsymbol{\alpha}_i$ is not exploited in Eq. (8).

Recall that in computing the NARM matrix \boldsymbol{S} , for each patch \mathbf{x}_i we have identified a set of nonlocal neighbors \mathbf{x}_i^j of it. Since in general \mathbf{x}_i can be well approximated by its nonlocal neighbors \mathbf{x}_i^j as $\mathbf{x}_i \approx \sum_j \omega_i^j \mathbf{x}_i^j$, the coding vectors of \mathbf{x}_i^j , denoted by $\boldsymbol{\alpha}_i^j$, should also be able to approximate $\boldsymbol{\alpha}_i$. Thus, $\boldsymbol{\alpha}_i$ should be close to the weighted average of $\boldsymbol{\alpha}_i^j$, i.e., $\|\boldsymbol{\alpha}_i - \sum_j \omega_i^j \boldsymbol{\alpha}_i^j\|_2$ should be small. Let

$$\boldsymbol{\alpha}_i^* = \sum_j \omega_i^j \boldsymbol{\alpha}_i^j. \quad (9)$$

We have the following nonlocal regularized SRM:

$$\hat{\boldsymbol{\alpha}} = \arg \min_{\boldsymbol{\alpha}} \left\{ \|\mathbf{y} - D\boldsymbol{S}\boldsymbol{\Psi}\boldsymbol{\alpha}\|_2^2 + \lambda \sum_{i=1}^N \|\boldsymbol{\alpha}_i\|_1 + \eta \sum_{i=1}^N \|\boldsymbol{\alpha}_i - \boldsymbol{\alpha}_i^*\|_2^2 \right\} \text{ s.t. } \mathbf{y} = D\boldsymbol{\Psi}\boldsymbol{\alpha}, \quad (10)$$

where η is a constant. Compared with Eq. (8), the newly introduced nonlocal regularization term $\sum_{i=1}^N \|\alpha_i - \alpha_i^*\|_2^2$ in Eq. (10) could make the sparse coding more accurate by exploiting the nonlocal redundancy. In addition, it has been shown in [21, 41] that the re-weighted l_p -norm ($p=1$ or 2) can enhance the sparsity and lead to better results. Therefore, we extend the proposed nonlocal regularized SRM in Eq. (10) to re-weighted nonlocal regularized SRM:

$$\hat{\alpha} = \arg \min_{\alpha} \left\{ \|\mathbf{y} - \mathbf{D}\mathbf{S}\Psi\alpha\|_2^2 + \sum_{i=1}^N \sum_j^r \lambda_{i,j} |\alpha_{i,j}| + \sum_{i=1}^N \sum_j^r \eta_{i,j} (\alpha_{i,j} - \alpha_{i,j}^*)^2 \right\} \text{ s.t. } \mathbf{y} = \mathbf{D}\Psi\alpha, \quad (11)$$

where $\alpha_{i,j}$ and $\alpha_{i,j}^*$ denote the j^{th} element of vector α_i and α_i^* , respectively, and $\lambda_{i,j}$ and $\eta_{i,j}$ are the weights assigned to the sparsity and nonlocal terms, respectively.

We re-write Eq. (11) as

$$\hat{\alpha} = \arg \min_{\alpha} \left\{ \|\mathbf{y} - \mathbf{D}\mathbf{S}\Psi\alpha\|_2^2 + \sum_{i=1}^N \|\lambda_i \alpha_i\|_1 + \sum_{i=1}^N \|\boldsymbol{\eta}_i (\alpha_i - \alpha_i^*)\|_2^2 \right\} \text{ s.t. } \mathbf{y} = \mathbf{D}\Psi\alpha, \quad (12)$$

where λ_i and $\boldsymbol{\eta}_i$ are diagonal weighting matrices whose diagonal elements are $\lambda_{i,j}$ and $\eta_{i,j}$, respectively. In the literature of variational image restoration [40], it has been shown that the regularization parameter should be inversely proportional to the signal-to-noise-ratio (SNR). In the iteratively re-weighted l_p -norm minimization [41, 21], it is suggested that the weight assigned to coding coefficient α should be set as $|\alpha|^{p-2}$. Therefore, for each element of the coding vector α_i , denoted by $\alpha_{i,j}$, we set adaptively the associated weights $\lambda_{i,j}$ and $\eta_{i,j}$ as

$$\lambda_{i,j} = \frac{c_1}{|\alpha_{i,j}^{(l)}| + \varepsilon}, \quad \eta_{i,j} = \frac{c_2}{(\alpha_{i,j}^{(l)} - \alpha_{i,j}^*)^2 + \varepsilon}, \quad (13)$$

where c_1 and c_2 are two predefined positive constants, $\alpha_{i,j}^{(l)}$ denotes the estimate of $\alpha_{i,j}$ obtained in the l^{th} iteration, and ε is a small positive number to increase the stability of Eq. (13). The detailed algorithm to minimize the model in Eq. (12) will be given in Section IV.

B. The selection of dictionary

In the proposed NARM based SRM in Eq. (12), the analytically designed dictionary, such as DCT, wavelet and curvelet dictionaries, can be used to span the sparse domain. However, recent studies have shown that

dictionaries learned from example images can lead to better image restoration results [23, 30, 39]. Moreover, in [7] a set of compact PCA sub-dictionaries, instead of one single over-complete dictionary, are learned from several natural images. Then for a given patch \mathbf{x}_i , a sub-dictionary is adaptively selected to code it. Such an adaptive sparse domain selection (ASDS) strategy has shown very promising results [7].

In this paper, we adopt this ASDS strategy to better characterize the image local structures. Similar to [7], for each patch \mathbf{x}_i we adaptively select one compact PCA dictionary; different from [7], where the PCA dictionaries are pre-learned from example high quality images, in this paper we learn online the sub-dictionaries. With an initial estimate of \mathbf{x} , we cluster the local patches into K clusters, denoted by \mathbf{Z}_k , $k=1, 2, \dots, K$. For each cluster \mathbf{Z}_k , since the patches within it are similar to each other, it is not necessary to learn an over-complete dictionary, and we simply use PCA to learn a compact dictionary Ψ_k for cluster \mathbf{Z}_k . Those sub-dictionaries actually form a big over-complete dictionary $\Psi=[\Psi_1, \Psi_2, \dots, \Psi_k]$ for the whole space of image patches. For a given patch \mathbf{x}_i to be coded, we first check which cluster is the closest one to it, and then select the sub-dictionary associated with this cluster, say Ψ_k , to code \mathbf{x}_i . This actually enforces the coding coefficients of \mathbf{x}_i over the remaining sub-dictionaries to be $\mathbf{0}$, leading to a very sparse representation of \mathbf{x}_i . Furthermore, by using the adaptive PCA dictionary, we only need to use a few major eigenvectors of Ψ_k to code \mathbf{x}_i , and this could further sparsify the coding coefficients of \mathbf{x}_i . For more details about the clustering and PCA based compact dictionary learning, please refer to [7]. In implementation, we update the PCA dictionaries in several iterations to reduce computational cost.

4. The Interpolation Algorithm

Given the current estimate of \mathbf{x} , the NARM matrix \mathbf{S} , the α_i^* and the weighting matrixes, λ_i and η_i in Eq. (12) can be calculated for the next iteration of minimization. After updating the PCA dictionaries Ψ (or we can update them in several iterations), we can then update \mathbf{x} by minimizing the energy function in Eq. (12). Consequently, the updated estimate of \mathbf{x} is used to update \mathbf{S} , Ψ , α_i^* , λ_i and η_i , which are in turn used to improve the reconstruction of \mathbf{x} . Such an iterative minimization process terminates until some stopping criterion is met.

A. Algorithm

In this paper, we adopt the variable splitting technique [46] to solve the constrained minimization in Eq. (12).

By introducing a quadratic term, we convert the objective function in Eq. (12) to:

$$(\hat{\mathbf{x}}, \{\hat{\boldsymbol{\alpha}}_i\}) = \arg \min_{\mathbf{x}, \{\boldsymbol{\alpha}_i\}} \left\{ \|\mathbf{y} - \mathbf{DSx}\|_2^2 + \beta \sum_{i=1}^N \|\mathbf{R}_i \mathbf{x} - \boldsymbol{\Psi} \boldsymbol{\alpha}_i\|_2^2 + \sum_{i=1}^N \|\boldsymbol{\lambda}_i \boldsymbol{\alpha}_i\|_1 + \sum_{i=1}^N \|\boldsymbol{\eta}_i (\boldsymbol{\alpha}_i - \boldsymbol{\alpha}_i^*)\|_2^2 \right\}, \text{ s.t. } \mathbf{y} = \mathbf{Dx}, \quad (14)$$

where \mathbf{R}_i is the matrix extracting a local patch from \mathbf{x} at position i . With a large enough parameter β , $\mathbf{R}_i \mathbf{x}$ approaches to $\boldsymbol{\Psi} \boldsymbol{\alpha}_i$, and the above objective function approaches to Eq. (12).

As described in Sec. III-B, instead of using a single over-complete dictionary, we use a set of sub-dictionaries to adaptively characterize image local structures. We cluster the image patches into K clusters and learn a PCA dictionary for each cluster. Then the learned dictionary $\boldsymbol{\Psi}_k$ for cluster k is assigned to the patches falling into this cluster. With the adaptive local dictionary, the objective function can be written as:

$$(\hat{\mathbf{x}}, \{\hat{\boldsymbol{\alpha}}_i\}) = \arg \min_{\mathbf{x}, \{\boldsymbol{\alpha}_i\}} \left\{ \|\mathbf{y} - \mathbf{DSx}\|_2^2 + \beta \sum_{k=1}^K \sum_{i \in C_k} \|\mathbf{R}_i \mathbf{x} - \boldsymbol{\Psi}_k \boldsymbol{\alpha}_i\|_2^2 + \sum_{i=1}^N \|\boldsymbol{\lambda}_i \boldsymbol{\alpha}_i\|_1 + \sum_{i=1}^N \|\boldsymbol{\eta}_i (\boldsymbol{\alpha}_i - \boldsymbol{\alpha}_i^*)\|_2^2 \right\}, \text{ s.t. } \mathbf{y} = \mathbf{Dx}, \quad (15)$$

where C_k is the set of indices of patches within cluster k . The above minimization problem can be solved by alternatively optimizing \mathbf{x} and $\{\boldsymbol{\alpha}_i\}$. For a set of fixed sparse codes $\{\boldsymbol{\alpha}_i\}$, \mathbf{x} can be optimized by minimizing

$$\hat{\mathbf{x}} = \arg \min_{\mathbf{x}} \left\{ \|\mathbf{y} - \mathbf{DSx}\|_2^2 + \beta \sum_{k=1}^K \sum_{i \in S_k} \|\mathbf{R}_i \mathbf{x} - \boldsymbol{\Psi}_k \boldsymbol{\alpha}_i\|_2^2 \right\}, \text{ s.t. } \mathbf{y} = \mathbf{Dx}, \quad (16)$$

and for a fixed \mathbf{x} , the set of sparse codes $\{\boldsymbol{\alpha}_i\}$ can be solved by minimizing

$$\{\hat{\boldsymbol{\alpha}}_i\} = \arg \min_{\{\boldsymbol{\alpha}_i\}} \left\{ \beta \sum_{k=1}^K \sum_{i \in S_k} \|\mathbf{R}_i \mathbf{x} - \boldsymbol{\Psi}_k \boldsymbol{\alpha}_i\|_2^2 + \sum_{i=1}^N \|\boldsymbol{\lambda}_i \boldsymbol{\alpha}_i\|_1 + \sum_{i=1}^N \|\boldsymbol{\eta}_i (\boldsymbol{\alpha}_i - \boldsymbol{\alpha}_i^*)\|_2^2 \right\}. \quad (17)$$

The above optimization processes can be iterated until converged. During the iterative process, we gradually increase the value of β such that Eq. (14) can well approximate Eq. (12).

One major advantage of variable splitting technique lies in that it can split a difficult problem into two sub-problems that are much easier to solve. For Eq. (16), we employ the Augmented Lagrange Multiplier (ALM) algorithm [47-48] to solve it. With ALM, we have the following Lagrangian function of Eq. (16):

$$L(\mathbf{x}, \mathbf{Z}, \mu) = \|\mathbf{y} - \mathbf{DSx}\|_2^2 + \beta \sum_{k=1}^K \sum_{i \in C_k} \|\mathbf{R}_i \mathbf{x} - \boldsymbol{\Psi}_k \boldsymbol{\alpha}_i\|_2^2 + \langle \mathbf{Z}, \mathbf{y} - \mathbf{Dx} \rangle + \mu \|\mathbf{y} - \mathbf{Dx}\|_2^2, \quad (18)$$

where $\langle \cdot, \cdot \rangle$ denotes the inner product, \mathbf{Z} is the Lagrangian multiplier, and μ is a positive scalar. Then the

optimization problem of Eq. (16) can be solved by the ALM method, which consists of the following iterations [47]

$$\mathbf{x}^{(l+1)} = \arg \min_{\mathbf{x}} L(\mathbf{x}, \mathbf{Z}^{(l)}, \mu^{(l)}), \quad (19)$$

$$\mathbf{Z}^{(l+1)} = \mathbf{Z}^{(l)} + \mu^{(l)}(\mathbf{y} - \mathbf{D}\mathbf{x}^{(l+1)}), \quad (20)$$

$$\mu^{(l+1)} = \tau \cdot \mu^{(l)}, \quad (21)$$

where τ ($\tau > 1$) is a constant. For fixed $\mathbf{Z}^{(l)}$ and $\mu^{(l)}$, we solve Eq. (19) for \mathbf{x} by taking $\partial L(\mathbf{x}, \mathbf{Z}^{(l)}, \mu^{(l)}) / \partial \mathbf{x} = 0$,

leading to the following equation:

$$\mathbf{x}^{(l+1)} = \left[(\mathbf{DS})^T \mathbf{DS} + \beta \sum_{i=1}^N \mathbf{R}_i^T \mathbf{R}_i + \mu^{(l)} \mathbf{D}^T \mathbf{D} \right]^{-1} \left[(\mathbf{DS})^T \mathbf{y} + \beta \sum_{i=1}^N \mathbf{R}_i^T \mathbf{R}_i (\Psi_k \alpha_i) + \mathbf{D}^T \mathbf{Z}^{(l)} / 2 + \mu^{(l)} \mathbf{D}^T \mathbf{y} \right]. \quad (22)$$

Since the matrix to be inverted in the right side of Eq. (22) is very large, we use the conjugate gradient algorithm to compute \mathbf{x} . With the updated estimate of \mathbf{x} , \mathbf{Z} and μ can be easily updated [47]. The procedures can be iterated until converged.

For a given \mathbf{x} , the minimization of Eq. (17) is a typical patch-based sparse coding problem. For each patch i , we solve the following sparse coding problem

$$\hat{\alpha}_i = \arg \min_{\alpha} \left\{ \beta \|\mathbf{R}_i \mathbf{x} - \Psi_k \alpha\|_2^2 + \|\lambda_i \alpha\|_1 + \|\eta_i (\alpha - \alpha^*)\|_2^2 \right\}. \quad (23)$$

To solve the above nonlocal regularized sparse coding problem, we extend the iterative shrinkage method in [16, 13] from handling one l_1 -norm constraint to handling mixed l_1 and l_2 -norm constraints. The closed-form shrinkage function can be derived as

$$\hat{\alpha}_{i,j} = \text{Soft}(v_{i,j}) = \begin{cases} 0, & |v_{i,j}| \leq \frac{\tau_{1,j}}{2\tau_{2,j} + 1} \\ v_{i,j} - \text{sign}(v_{i,j}) \frac{\tau_{1,j}}{2\tau_{2,j} + 1}, & \text{otherwise} \end{cases}, \quad (24)$$

where $v_{i,j} = \gamma_{i,j} / (2\tau_{2,j} + 1)$ with $\gamma_i = 2\eta_i \alpha_i^* / \beta + \Psi_k^T \mathbf{R}_i \mathbf{x}_i$, $\tau_{1,j} = \lambda_{i,j} / \beta$ and $\tau_{2,j} = \eta_{i,j} / \beta$.

The proposed NARM based SRM algorithm for image interpolation is summarized in **Algorithm 1**.

Algorithm 1: NARM based SRM for Image Interpolation

1. Initialization:
 - (a) Initialize \mathbf{x} using the bi-cubic interpolator;
 - (b) Compute $\{\Psi_k\}$ via patch clustering and PCA, and compute NARM matrix \mathbf{S} and α_i^* ;
 - (c) Set $\mathbf{Z}^{(0)}=\mathbf{0}$, $\mu^{(0)}>0$, $\beta>0$, $\tau>1$, $\rho>1$, $\lambda>0$, and $\eta>0$;
 2. Outer loop, $j=1,2,\dots,J$
 - (a) Solving Eq. (17) for $\{\alpha_i\}$: compute $\alpha_i^{(j+1)}$ using Eq. (24) ;
 - (b) Inner loop, $l=1,2,\dots,L$
 - i. Solving Eq. (19) for $\mathbf{x}^{(l+1)}$ by using the conjugate gradient algorithm;
 - ii. $\mathbf{Z}^{(l+1)} = \mathbf{Z}^{(l)} + \mu^{(l)}(\mathbf{y} - \mathbf{D}\mathbf{x}^{(l+1)})$;
 - iii. $\mu^{(l+1)} = \tau \cdot \mu^{(l)}$;
 - (c) If $\text{mod}(j, J_0)$, update $\{\Psi_k\}$, \mathbf{S} , α_i^* , λ_{ij} and η_{ij} via Eq. (13);
 - (d) $\beta^{(j+1)} = \rho \cdot \beta^{(j)}$;
 - (e) Go back to step 2-(a) till the algorithm converges or the maximum number of iteration is reached.
 3. Output the interpolated image \mathbf{x} .
-

In the outer loop of **Algorithm 1**, we compute the set of sparse codes $\{\alpha_i\}$ using the current estimate of original image \mathbf{x} , and update $\{\Psi_k\}$, \mathbf{S} , α_i^* , λ_{ij} and η_{ij} in every J_0 iterations ($J_0=15$ in our implementation) to save the computational cost. The inner loop that solves the constrained minimization of Eq. (16) follows the standard procedures of ALM algorithm [47]. Our numerical tests show that **Algorithm 1** converges even with $L=1$, and thus we only need to execute the conjugate gradient algorithm (to solve Eq. (22)) once in each outer loop, saving much the cost. Our experiments show that **Algorithm 1** usually converges in 50 iterations.

B. Computational complexity

The computational cost of **Algorithm 1** mainly comes from four sources, i.e., the clustering-based PCA sub-dictionary learning in Step 1(b) and Step 2(c), the NARM computing in Step 1(b) and Step 2(c), the patch-based sparse coding in Step 2(a), and the conjugate gradient minimization in Step 2(b).

The patch clustering needs $O(u \cdot K \cdot q \cdot n)$ operations, where u ($u=12$ in our implementation) is the number of iterations in K -means clustering, q is the total number of patches extracted for clustering (to save computational cost we exclude those smooth patches whose intensity variances are smaller than a threshold from clustering), and n is the length of the patch vector. The computation of PCA sub-dictionaries needs $O(K \cdot (m^2 \cdot n^2 + n^3))$ operations, where we assume that each cluster has m patch samples in average. Thus, the

clustering-based PCA sub-dictionary learning needs $O(T(u \cdot K \cdot q \cdot n + K \cdot (m^2 \cdot n^2 + n^3)))$ operations in total, where T counts the times of PCA sub-dictionary update in the whole algorithm implementation.

The NARM modeling involves N_L times of K -nearest neighbor search for nonlocal similar patches and conjugate gradient minimization for solving Eq. (4), where N_L is the number of LR pixels. Thus, this process needs $O(N_L \cdot (s^2 \cdot n + t_1 \cdot p^2))$ operations in total, where s is the width of searching window, t_1 is the number of iterations in conjugate gradient minimization, and p is the number of nonlocal samples used for NARM modeling. The sparse coding by Eq. (23) needs $O(N \cdot (2 \cdot n^2 + p \cdot n))$ operations, where N is the total number of patches extracted for sparse coding. The major computational cost in the conjugate gradient algorithm for computing Eq. (22) comes from the matrix-vector multiplication, i.e., multiplying \mathbf{G} with vector \mathbf{x} , where $\mathbf{G} = (\mathbf{DS})^T \mathbf{DS} + \beta \sum_{i=1}^N \mathbf{R}_i^T \mathbf{R}_i + \mu \mathbf{D}^T \mathbf{D}$. Since \mathbf{DS} and $\mathbf{D}^T \mathbf{D}$ are sparse and $\sum_{i=1}^N \mathbf{R}_i^T \mathbf{R}_i$ can be pre-computed, the matrix-vector multiplication can be executed very fast. In general, the conjugate gradient minimization converges in less than κ iterations, where κ is the condition number of matrix \mathbf{G} . Thus, the cost of conjugate gradient minimization is $O(J \cdot \kappa \cdot (N_L \cdot N \cdot (p+1) + N \cdot n))$, where J is the total number of outer loop iterations in **Algorithm 1**.

Based on our experience, it costs about 2~4 minutes to interpolate an LR image of size 128×128 to an HR image of size 256×256 by running **Algorithm 1** on an Intel Core2Duo i7 2.67G laptop PC under Matlab R2011a environment. The running time is shorter for smooth images than non-smooth images, since the many smooth patches will be excluded from patch clustering. For example, the running time of image *Foreman* is about 2.5 minutes, while the running time of image *Fence* is 3.6 minutes. The number of patches extracted for sparse coding also influences the running time. For example, if the patches are extracted in every two pixels along the horizontal and vertical directions, the running time of *Foreman* is about 2.5 minutes. If the patches are extracted at every pixel, the running time increases to 3.0 minutes with a slight increase of PSNR (about 0.1 dB).

5. Experimental Results



Figure 5: Test images. From left to right and top to bottom: *Lena*, *House*, *Foreman*, *Leaves*, *Peppers*, *Butterfly*, *Girl*, *Fence*, *Parthenon*, and *Starfish*.

In this section we evaluate the proposed interpolation method. Refer to Fig. 5, ten widely used test images in the literature are employed in the experiments. In our implementation, the patch size is set to 5×5 . In all experiments, the LR images are generated by down-sampling directly the original HR images with a factor of s ($s = 2$ or 3). For color images, we only apply the proposed method to the luminance channel since human visual system is more sensitive to luminance changes, and apply the bi-cubic interpolator to chromatic channels. To evaluate the quality of interpolated images, the PSNR and two perceptual quality metrics, SSIM [42] and FSIM [43], are computed to compare the competing interpolation algorithms. For color images, we only report the PSNR, SSIM and FSIM measures for the luminance channel. All the experimental results can be downloaded from <http://www.comp.polyu.edu.hk/~cslzhang/NARM.htm>.

To more comprehensively evaluate the effectiveness of the proposed NARM modeling as well as the nonlocal regularization, we first conduct experiments with four variants of the proposed method. We then compare the proposed method with other state-of-the-art image interpolation methods. At last, we discuss the parameter selection of the algorithm.

A. Effectiveness of NARM and nonlocal regularization

To demonstrate the effectiveness of the proposed NARM and sparsity regularization, we implement four variants of the proposed interpolation algorithm. First, let's remove the NARM from the data fidelity term. There are two variants in this case. The first one, denoted by "SRM", solves the standard SRM minimization:

$$\hat{\alpha} = \arg \min_{\alpha} \left\{ \|y - D\Psi\alpha\|_2^2 + \sum_{i=1}^N \|\lambda_i \alpha_i\|_1 \right\} \text{ s.t. } y = D\Psi\alpha. \quad (25)$$

The original image is reconstructed by $\hat{x} = \Psi\hat{\alpha}$. The second variant, denoted by "SRM-NL", solves the SRM minimization with nonlocal regularization:

$$\hat{\alpha} = \arg \min_{\alpha} \left\{ \|y - D\Psi\alpha\|_2^2 + \sum_{i=1}^N \|\lambda_i \alpha_i\|_1 + \sum_{i=1}^N \|\eta_i (\alpha_i - \alpha_i^*)\|_2^2 \right\} \text{ s.t. } y = D\Psi\alpha. \quad (26)$$

By incorporating NARM into the data fidelity term, there are other two variants. Let's denote by "NARM-SRM" the NARM based SRM which solves the following minimization problem:

$$\hat{\alpha} = \arg \min_{\alpha} \left\{ \|y - DS\Psi\alpha\|_2^2 + \sum_{i=1}^N \|\lambda_i \alpha_i\|_1 \right\} \text{ s.t. } y = D\Psi\alpha. \quad (27)$$

The last variant, denoted by "NARM-SRM-NL", solves the following minimization problem with nonlocal regularization:

$$\hat{\alpha} = \arg \min_{\alpha} \left\{ \|y - DS\Psi\alpha\|_2^2 + \sum_{i=1}^N \|\lambda_i \alpha_i\|_1 + \sum_{i=1}^N \|\eta_i (\alpha_i - \alpha_i^*)\|_2^2 \right\} \text{ s.t. } y = D\Psi\alpha. \quad (28)$$

The variant NARM-SRM-NL can be implemented by **Algorithm 1**, while the other three variants can be implemented by modifying slightly **Algorithm 1**. Applying the four variants to the test images in Fig. 5, we show the PSNR results in Table 1. One can see that both the NARM based data fidelity constraint $\|y - DSx\|_2^2$ and the nonlocal regularization $\|\alpha_i - \alpha_i^*\|_2^2$ can much improve the performance of standard SRM method. The SRM-NL method slightly outperforms NARM-SRM. However, by combining $\|y - DSx\|_2^2$ and $\|\alpha_i - \alpha_i^*\|_2^2$, the NARM-SRM-NL method achieves the best interpolation result. Some reconstructed HR images of the proposed methods are shown in Figs. 6~7.

Table 1: PSNR (dB) results of the four variants of the proposed method.

Scaling factor $s=2$											
<i>Images</i>	<i>Lena</i>	<i>House</i>	<i>F.man</i>	<i>Leaves</i>	<i>Cam.</i>	<i>Butterfly</i>	<i>Girl</i>	<i>Fence</i>	<i>Parth.</i>	<i>Starfish</i>	<i>Average</i>
SRM	34.27	32.64	36.55	27.62	25.60	28.75	34.20	24.64	27.33	31.02	30.26
NARM-SRM	34.66	33.19	38.26	28.71	25.43	29.45	34.29	24.60	27.14	31.78	30.75
SRM-NL	34.89	33.29	38.32	29.15	25.88	29.81	34.11	24.72	27.42	31.21	30.88
NARM-SRM-NL	35.04	33.45	38.60	29.77	25.92	30.35	34.28	24.70	27.26	31.73	31.11
Scaling factor $s=3$											
SRM	30.23	29.21	32.50	22.15	22.55	23.92	31.20	20.71	24.65	26.47	26.36
NARM-SRM	30.43	29.46	33.40	22.75	22.44	24.65	31.58	20.61	24.56	26.59	26.65
SRM-NL	31.03	29.76	34.08	23.03	22.76	25.01	30.50	20.61	24.77	26.71	26.83
NARM-SRM-NL	31.18	29.76	34.60	23.32	22.60	25.48	31.95	20.49	24.66	26.78	27.08



(a) SRM (b) NARM-SRM (c) SRM-NL (d) NARM-SRM-NL

Figure 6: Reconstructed images on *Starfish* (zooming factor $s=2$). (a) PSNR=31.02 dB; (b) PSNR=31.78 dB; (c) PSNR=31.21 dB; (d) PSNR=31.73 dB.

(a) SRM (b) NARM-SRM (c) SRM-NL (d) NARM-SRM-NL

Figure 7: Reconstructed images on *Butterfly* (zooming factor $s=2$). (a) PSNR=28.75 dB; (b) PSNR=29.45 dB; (c) PSNR=29.81 dB; (d) PSNR=30.35 dB.

B. Effectiveness of the local PCA dictionaries

To verify the effectiveness of the local PCA dictionaries for image interpolation, we replace the local PCA dictionaries with the DCT dictionary in the SRM and NARM-SRM-NL algorithms, and denote by “DCT-SRM” and “DCT-NARM-SRM-NL”, respectively, these two DCT dictionary based implementations. The results are shown in Table 2 and Figs. 8~9. One can see that the SRM outperforms DCT-SRM, while

NARM-SRM-NL outperforms DCT-NARM-SRM-NL, in term of both PSNR and visual quality. This validates that the local PCA dictionaries are better than DCT dictionary in reconstructing fine scale edges.

Table 2: Comparison of the interpolation results (dB) by DCT and local PCA dictionaries.

Scaling factor $s=2$											
Images	<i>Lena</i>	<i>House</i>	<i>F.man</i>	<i>Leaves</i>	<i>Cam.</i>	<i>Butterfly</i>	<i>Girl</i>	<i>Fence</i>	<i>Parth.</i>	<i>Starfish</i>	<i>Average</i>
DCT-SRM	33.63	32.85	35.93	27.14	25.42	28.14	33.67	24.55	27.25	30.09	29.87
SRM	34.27	32.64	36.55	27.62	25.60	28.75	34.20	24.64	27.33	31.02	30.26
DCT-NARM-SRM-NL	34.90	32.92	38.64	29.25	25.69	30.00	34.21	24.62	27.15	31.42	30.88
NARM-SRM-NL	35.04	33.45	38.60	29.77	25.92	30.35	34.28	24.70	27.26	31.73	31.11
Scaling factor $s=3$											
DCT-SRM	29.90	29.21	32.49	22.05	22.54	23.87	31.34	20.82	24.68	26.15	26.30
SRM	30.23	29.21	32.50	22.15	22.55	23.92	31.20	20.71	24.65	26.47	26.36
DCT-NARM-SRM-NL	31.10	29.11	34.61	23.15	22.67	25.41	31.61	20.57	24.61	26.71	26.96
NARM-SRM-NL	31.18	29.76	34.60	23.32	22.60	25.48	31.95	20.49	24.66	26.78	27.08

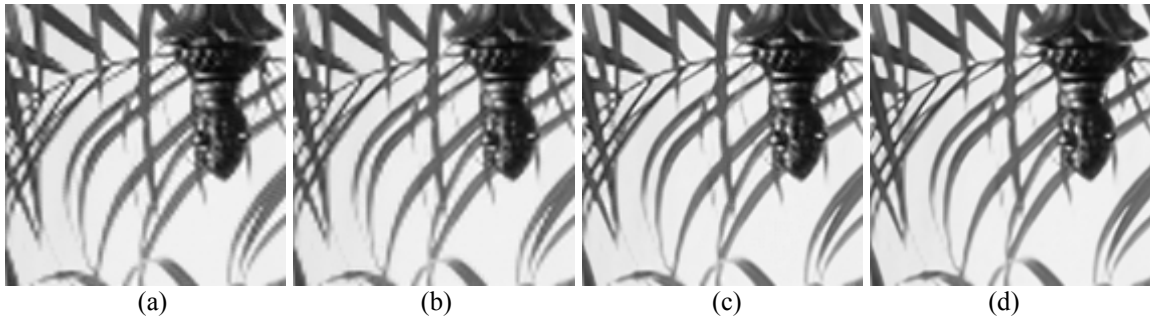


Figure 8: Reconstructed HR *Leaves* images (scaling factor $s=2$). (a) DCT-SRM (PSNR=27.14 dB); (b) SRM (PSNR=27.62 dB); (c) DCT-NARM-SRM-NL (PSNR=29.25 dB); (d) NARM-SRM-NL (PSNR=29.77 dB).

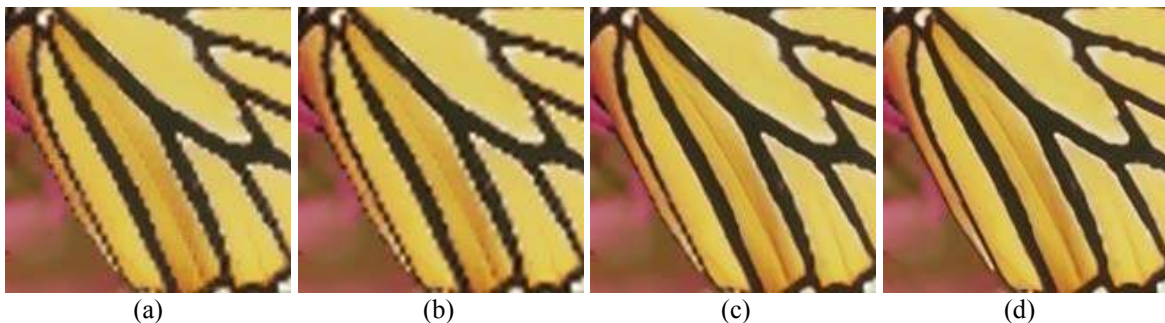


Figure 9: Reconstructed HR *Butterfly* images (scaling factor $s=3$). (a) DCT-SRM (PSNR=23.87 dB); (b) SRM (PSNR=23.92 dB); (c) DCT-NARM-SRM-NL (PSNR=25.41 dB); (d) NARM-SRM-NL (PSNR=25.48 dB).

C. Comparison with state-of-the-arts

We compare the proposed NARM-SRM-NL method with state-of-the-art image interpolation methods, including the NEDI method [3], the directional filtering and data fusion (DFDF) method [4], the soft-

decision and adaptive interpolator (SAI) [5], the recently developed sparsity-based ScSR [6] and SME (sparse mixing estimation) [8] methods³. The NEDI, DFDF and SAI are three well-known adaptive edge-directed image interpolation methods. The ScSR and SME are two recently developed sparsity-based image super-resolution/interpolation methods, which also show visually pleasant results.

We first report the results when the scaling factor $s=2$. The PSNR, SSIM and FSIM metrics by the competing methods on the ten test images are listed in Table 3. We can see that the proposed NARM based method achieves the highest PSNR, SSIM, and FSIM measures on almost all the test images. The PSNR gain of the proposed method over the second best method (i.e., the SAI method) can be up to 1.13 dB, and the average PSNR, SSIM and FSIM gains over SAI method are 0.68 dB, 0.0085, and 0.0057, respectively.

In Figs. 10~13, we show some cropped portions of the reconstructed HR images by the competing methods. From these figures, we can see that those edge based interpolation methods [3-5] reconstruct the image structures much better than the filtering based bi-cubic method. In particular, the SAI method [5] is very effective in preserving the large scale edges (e.g., the edges of starfish and butterfly in Fig. 10(g) and Fig. 11(g)). However, one problem of these edge-based methods is that they tend to generate artifacts in small scale edge structures (e.g., Fig. 12(g) and Fig. 13(g)). This is mainly because that it is difficult to accurately estimate the direction (or the local covariance) of the edges from the LR image. By using the sparsity prior, the sparsity-based methods ScSR [6] and SME [8] work better in handling those fine scale edges. However, they still cannot produce very sharp edges. Some ringing and jaggy artifacts can be clearly observed in the HR images reconstructed by ScSR and SME (e.g., Figs. 10(e)~(f) and Figs. 11(e)~(f)). By efficiently exploiting the nonlocal information and the sparsity prior, the proposed NARM based method can much improve the visual quality of the reconstructed HR images. It can be observed that NARM-SRM-NL produces not only sharp large-scale edges but also fine-scale image details.

Table 3: PSNR (dB), SSIM and FSIM results by different interpolation methods ($s=2$).

<i>Images</i>	Bi-cubic	NEDI [3]	DFDF [4]	ScSR [6]	SME [8]	SAI [5]	NARM-SRM-NL
<i>Lena</i>	33.91	33.76	33.89	33.70	34.53	34.68	35.01
	0.9140	0.9134	0.9122	0.9080	0.9178	0.9184	0.9238
	0.9872	0.9868	0.9867	0.9855	0.9881	0.9882	0.9893
<i>House</i>	32.15	31.67	32.57	31.78	33.15	32.84	33.52
	0.8772	0.8743	0.8775	0.8699	0.8811	0.8778	0.8841
	0.9404	0.9434	0.9478	0.9370	0.9515	0.9496	0.9567

³ We thank the authors for providing the source codes.

<i>Foreman</i>	35.56	35.90	36.81	35.68	37.17	37.68	38.64
	0.9491	0.9532	0.9541	0.9471	0.9554	0.9576	0.9581
	0.9652	0.9700	0.9712	0.9623	0.9724	0.9750	0.9754
<i>Leaves</i>	26.85	26.23	27.22	27.52	28.21	28.72	29.76
	0.9365	0.9403	0.9433	0.9460	0.9499	0.9575	0.9661
	0.9259	0.9429	0.9478	0.9293	0.9423	0.9591	0.9674
<i>Camera-man</i>	25.36	25.42	25.67	25.28	26.14	25.88	25.94
	0.8639	0.8626	0.8670	0.8611	0.8711	0.8709	0.8781
	0.9041	0.9059	0.9143	0.9031	0.9120	0.9177	0.9231
<i>Butterfly</i>	27.68	27.36	28.66	28.27	28.65	29.17	30.30
	0.9242	0.9321	0.9397	0.9315	0.9380	0.9468	0.9561
	0.9155	0.9284	0.9452	0.9151	0.9267	0.9466	0.9591
<i>Girl</i>	33.83	33.85	33.79	33.29	34.03	34.13	34.46
	0.8533	0.8570	0.8520	0.8411	0.8563	0.8588	0.8658
	0.9416	0.9412	0.9395	0.9335	0.9438	0.9444	0.9434
<i>Fence</i>	24.52	22.97	24.55	24.05	24.53	23.78	24.79
	0.7776	0.7586	0.7757	0.7645	0.7822	0.7704	0.7939
	0.8822	0.8825	0.8790	0.8869	0.8974	0.8921	0.9040
<i>Parthenon</i>	27.08	26.79	27.18	26.46	27.13	27.10	27.36
	0.8043	0.7883	0.8034	0.7813	0.7997	0.8014	0.8095
	0.8947	0.8911	0.8963	0.8886	0.9009	0.8980	0.9019
<i>Starfish</i>	30.22	29.36	30.07	30.35	30.35	30.76	31.72
	0.9169	0.8987	0.9118	0.9170	0.9165	0.9207	0.9299
	0.9522	0.9458	0.9541	0.9537	0.9523	0.9577	0.9648
<i>Average</i>	29.72	29.33	30.04	29.64	30.39	30.47	31.15
	0.8817	0.8779	0.8837	0.8768	0.8868	0.8880	0.8965
	0.9309	0.9338	0.9382	0.9295	0.9387	0.9428	0.9485

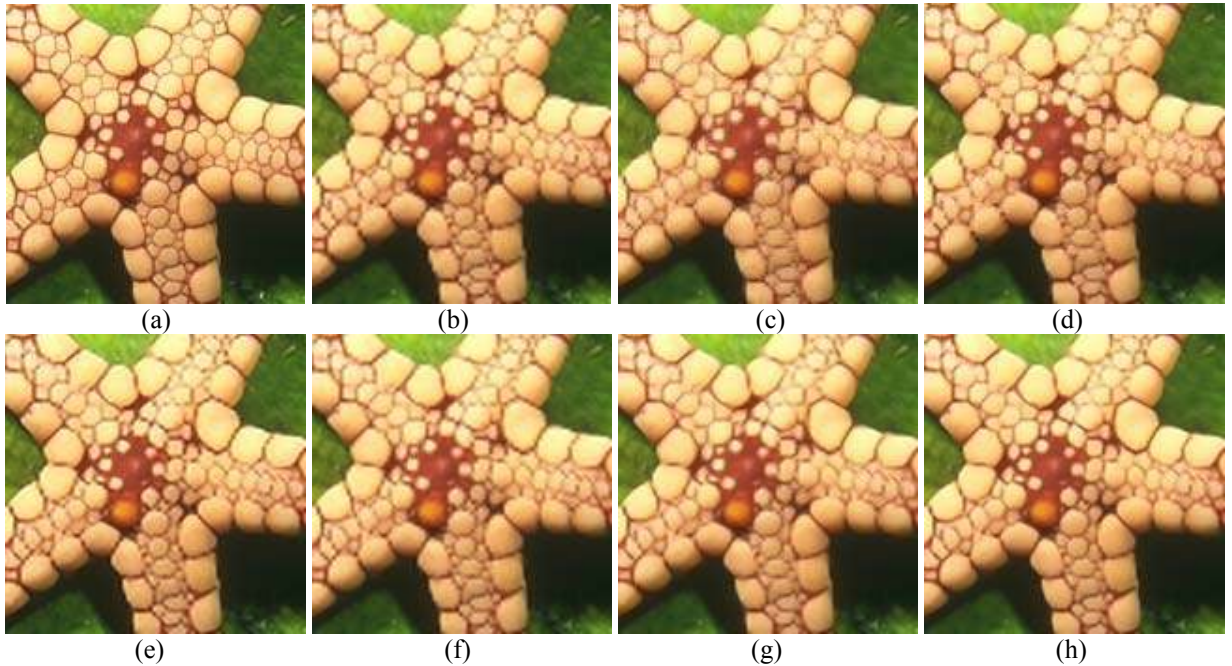


Figure 10: Reconstructed HR images (zooming factor $s=2$) of *Starfish* by different interpolation methods. (a) Original image; (b) Bi-cubic (PSNR=30.22dB, SSIM=0.9169, FSIM=0.9522); (c) NEDI [3] (PSNR=29.36dB, SSIM=0.8987, FSIM=0.9458); (d) DFDF [4] (PSNR=30.07dB, SSIM=0.9118, FSIM=0.9541); (e) ScSR [6] (PSNR=30.35dB, SSIM=0.9170, FSIM=0.9537); (f) SME [8] (PSNR=30.35dB, SSIM=0.9165, FSIM=0.9523); (g) SAI [5] (PSNR=30.76dB, SSIM=0.9207, FSIM=0.9577); (h) proposed NARM-SRM-NL (PSNR=31.72dB, SSIM=0.9299, FSIM=0.9648).

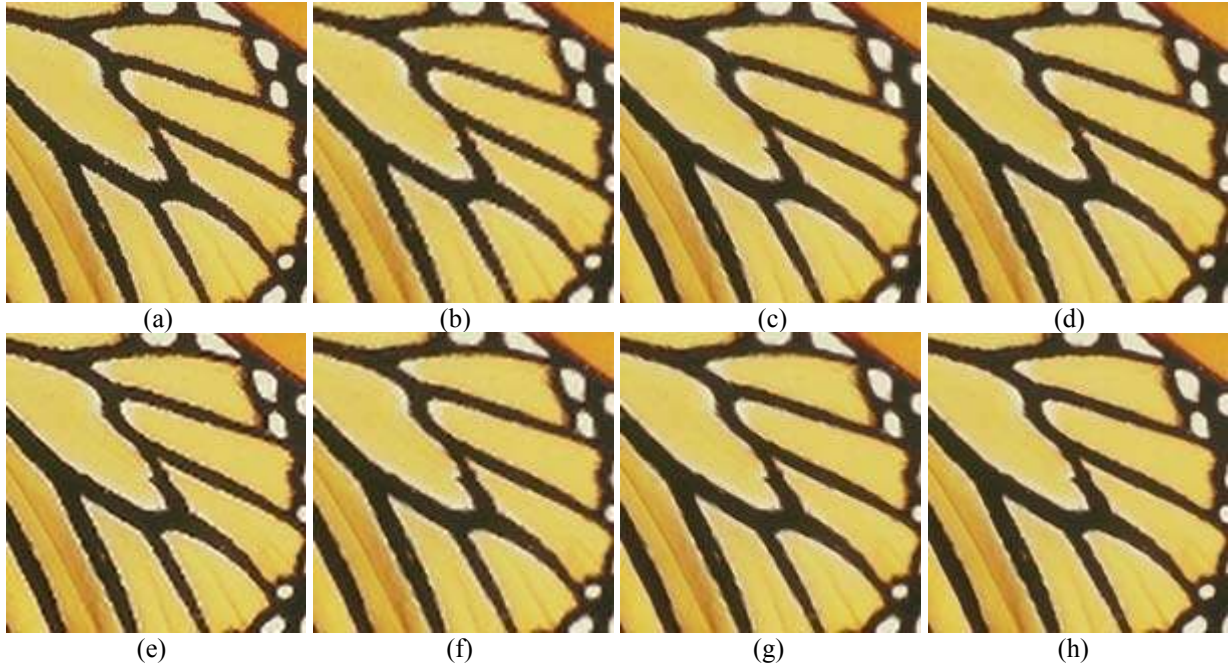


Figure 11: Reconstructed HR images (zooming factor $s=2$) of *Butterfly* by different interpolation methods. (a) Original image; (b) Bicubic (PSNR=27.68dB, SSIM=0.9242, FSIM=0.9155); (c) NEDI [3] (PSNR=27.36dB, SSIM=0.9321, FSIM=0.9284); (d) DFDF [4] (PSNR=28.66dB, SSIM=0.9397, FSIM=0.9452); (e) ScSR [6] (PSNR=28.27dB, SSIM=0.9315, FSIM=0.9151); (f) SME [8] (PSNR=28.65dB, SSIM=0.9380, FSIM=0.9267); (g) SAI [5] (PSNR=29.17dB, SSIM=0.9468, FSIM=0.9466); (h) proposed NARM-SRM-NL (PSNR=**30.30**dB, SSIM=**0.9561**, FSIM=**0.9591**).

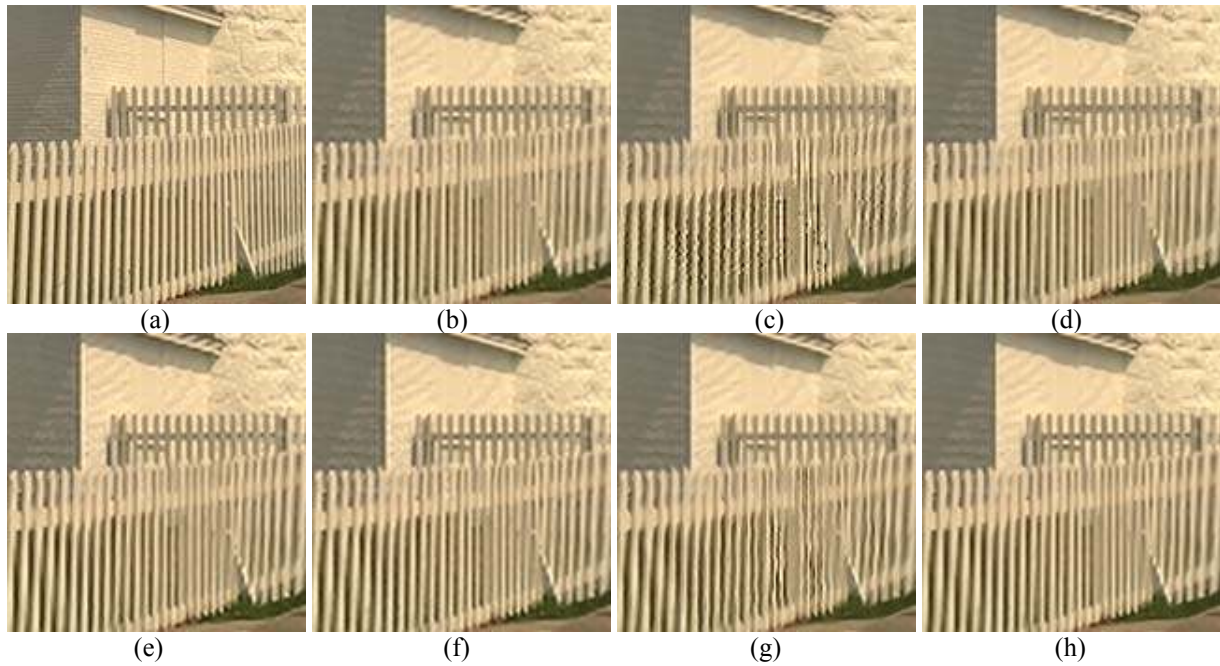


Figure 12: Reconstructed HR images (zooming factor $s=2$) of *Fence* by different interpolation methods. (a) Original image; (b) Bicubic (PSNR=24.52dB, SSIM=0.7776, FSIM=0.8822); (c) NEDI [3] (PSNR=22.97dB, SSIM=0.7586, FSIM=0.8825); (d) DFDF [4] (PSNR=24.55dB, SSIM=0.7757, FSIM=0.8790); (e) ScSR [6] (PSNR=24.02dB, SSIM=0.7645, FSIM=0.8869); (f) SME [8] (PSNR=24.53dB, SSIM=0.7822, FSIM=0.8974); (g) SAI [5] (PSNR=23.78dB, SSIM=0.7704, FSIM=0.8921); (h) proposed NARM-SRM-NL (PSNR=**24.79**dB, SSIM=**0.7939**, FSIM=**0.9040**).

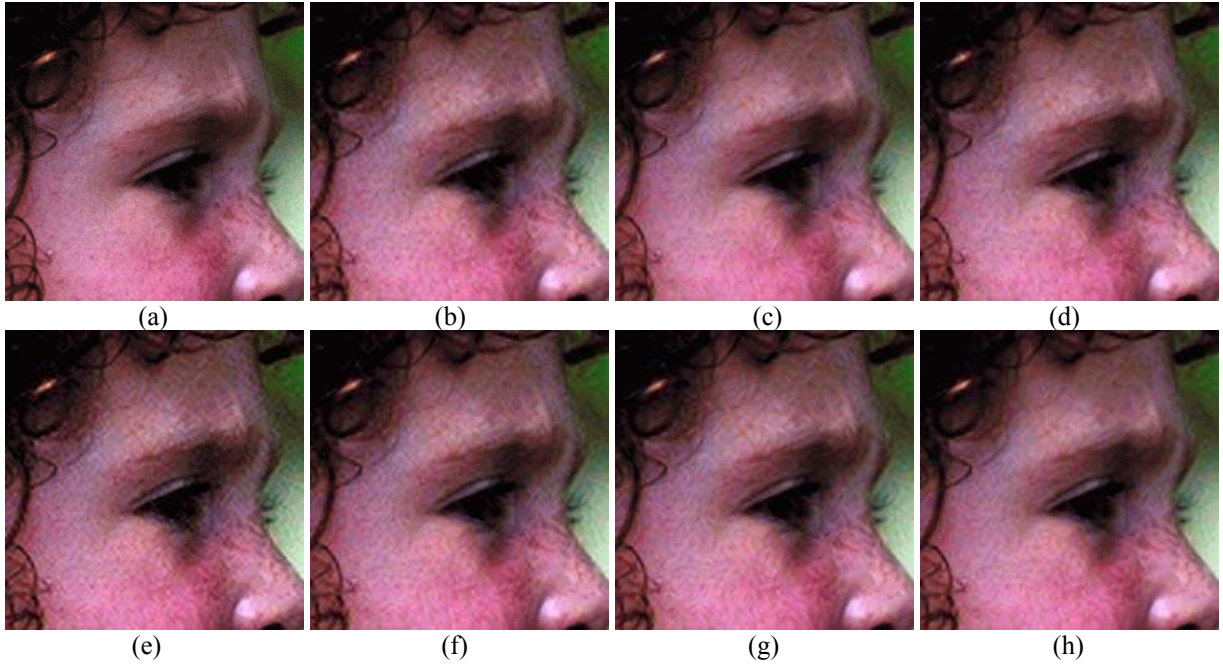


Figure 13: Reconstructed HR images (zooming factor $s=2$) of *Girl* by different interpolation methods. (a) Original image; (b) Bicubic (PSNR=33.83dB, SSIM=0.8533, FSIM=0.9416); (c) NEDI [3] (PSNR=33.85dB, SSIM=0.8570, FSIM=0.9412); (d) DFDF [4] (PSNR=33.79dB, SSIM=0.8520, FSIM=0.9395); (e) ScSR [6] (PSNR=33.29dB, SSIM=0.8411, FSIM=0.9335); (f) SME [8] (PSNR=34.03dB, SSIM=0.8563, FSIM=0.9438); (g) SAI [5] (PSNR=34.13dB, SSIM=0.8588, FSIM=0.9444); (h) proposed NARM-SRM-NL (PSNR=34.46dB, SSIM=0.8658, FSIM=0.9434).

The proposed NARM method is applicable to image interpolation with arbitrary integer scaling factor. In this section, we also conduct experiments with scaling factor $s=3$. Since the NEDI, DFDF, SAI and SME methods are designed for image interpolation with $s=2^n$, where n is an integer, we compare the NARM-SRM-NL algorithm with the bi-cubic algorithm and the ScSR algorithm. The PSNR, SSIM and FSIM results by the three methods are listed in Table 4, from which we can see that the proposed NARM-SRM-NL method outperforms ScSR and bi-cubic by a large margin. The average PSNR, SSIM and FSIM gains over ScSR method are up to 1.02 dB, 0.0254, and 0.0204, respectively. In Figs. 14~16, we show some cropped HR images reconstructed by the competing methods. One can see that ScSR produces sharper edges than bi-cubic; however, there are still many ringing artifacts around the edges. Clearly, the proposed NARM-SRM-NL generates the visually most pleasant HR images. The reconstructed edges by NARM-SRM-NL are much sharper than those by ScSR, while there are much fewer ringing and zipper artifacts.

Table 4: PSNR (dB), SSIM and FSIM results by different interpolation methods ($s=3$).

<i>Images</i>	<i>Lena</i>	<i>House</i>	<i>F.man</i>	<i>Leaves</i>	<i>Cam.</i>	<i>B.fly</i>	<i>Girl</i>	<i>Fence</i>	<i>Parth.</i>	<i>Starfish</i>	<i>Average</i>
Bi-cubic	30.14	28.66	32.07	21.85	22.36	23.48	31.45	20.64	24.28	26.18	26.11
	0.8550	0.8190	0.9079	0.8166	0.7686	0.8236	0.7772	0.5942	0.6790	0.8144	0.7855
	0.9651	0.8791	0.9285	0.8143	0.8208	0.8146	0.9040	0.7316	0.8146	0.8911	0.8564
ScSR [6]	30.00	28.53	32.29	21.93	22.21	23.84	31.10	20.38	24.06	26.08	26.04
	0.8472	0.8155	0.9073	0.8340	0.7673	0.8461	0.7653	0.5826	0.6710	0.8138	0.7850
	0.9609	0.8830	0.9289	0.8456	0.8286	0.8435	0.8966	0.7271	0.8138	0.8945	0.8622
NARM-SRM-NL	31.16	29.67	34.80	23.33	22.72	25.57	31.90	20.53	24.72	26.86	27.13
	0.8693	0.8371	0.9284	0.8827	0.7899	0.8993	0.7847	0.6005	0.6875	0.8293	0.8109
	0.9699	0.8911	0.9455	0.8955	0.8349	0.9028	0.8861	0.7349	0.8017	0.9060	0.8768

From Tables 3~4 and Figs. 10~16, we can see that the proposed NARM-SRM-NL interpolation method is very effective in reconstructing sharp edges when there are sufficient repetitive patterns in the image (e.g., images *Foreman*, *Butterfly* and *Lena*). This is because in such case we can accurately compute the NARM kernel \mathcal{S} , the nonlocal sparsity regularization term, and the local PCA dictionaries by exploiting the nonlocal redundancy. However, for some regions, e.g., the grass region in image *Cameraman* and the fence region in image *Fence*, after down-sampling severe aliasing effects will appear in the LR image, and it is rather challenging to robustly find enough similar patches to construct the NARM kernel \mathcal{S} , perform nonlocal regularization, and compute the local PCA dictionaries. As a result, the NARM method may fail to faithfully reconstruct those regions.

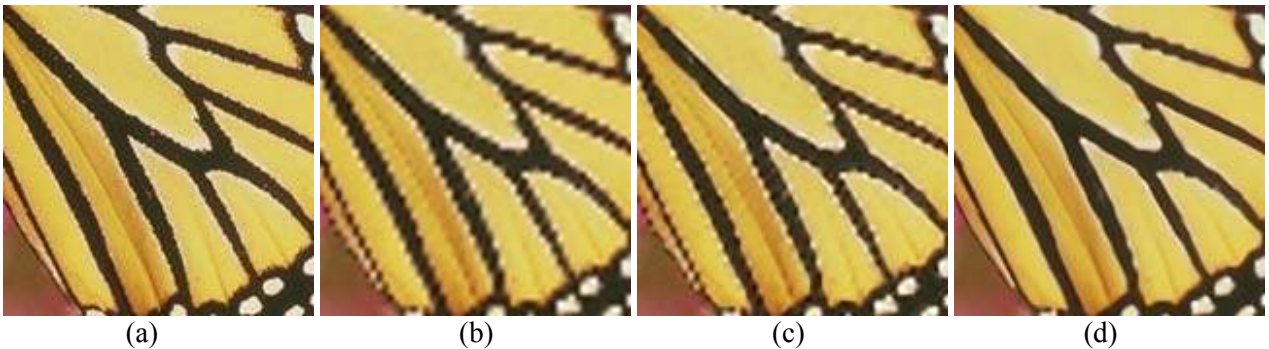


Figure 14: Reconstructed HR images (zooming factor $s=3$) of *Butterfly* by different interpolation methods. (a) Original image; (b) Bicubic (PSNR=23.48dB, SSIM=0.8236, FSIM=0.8146); (c) ScSR [6] (PSNR=23.84dB, SSIM=0.8461, FSIM=0.8435); (d) proposed NARM-SRM-NL (PSNR=25.57dB, SSIM=0.8993, FSIM=0.9028).

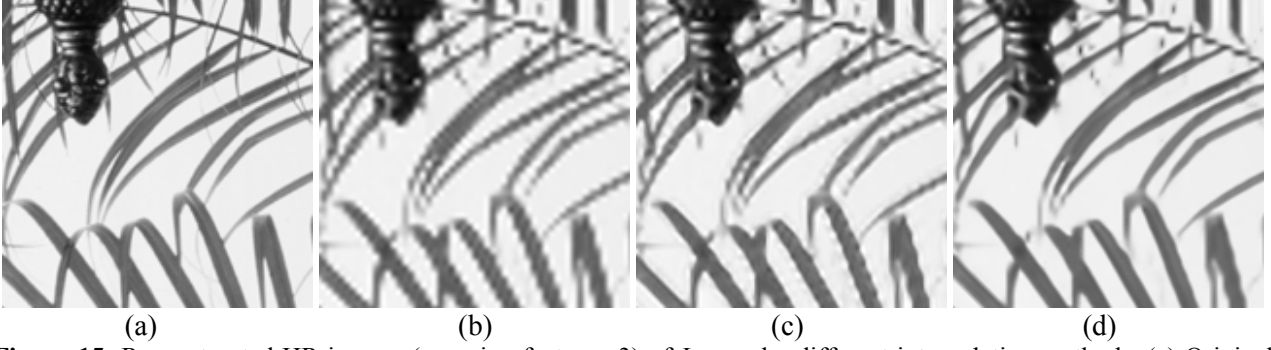


Figure 15: Reconstructed HR images (zooming factor $s=3$) of *Leaves* by different interpolation methods. (a) Original image; (b) Bicubic (PSNR=21.85dB, SSIM=0.8166, FSIM=0.8143); (c) ScSR [6] (PSNR=21.93dB, SSIM=0.8340, FSIM=0.8456); (d) proposed NARM-SRM-NL (PSNR=23.33dB, SSIM=0.8827, FSIM=0.8955).

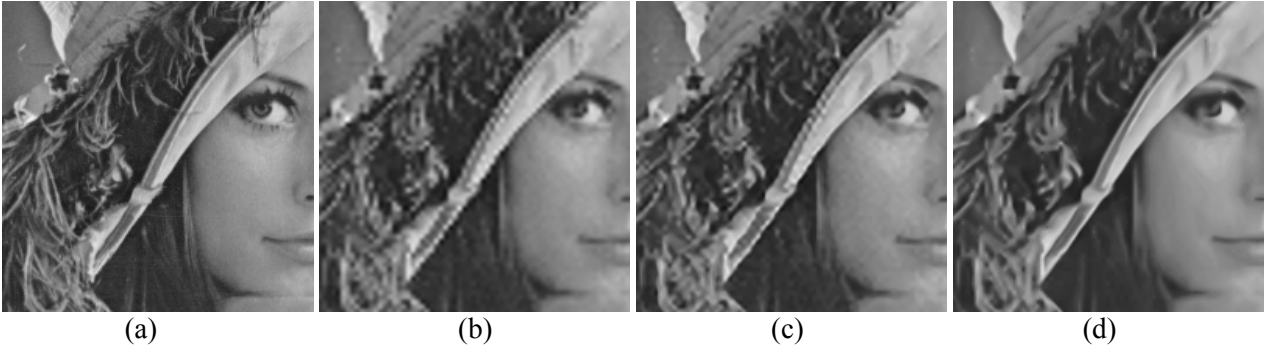


Figure 16: Reconstructed HR images (zooming factor $s=3$) of *Lena* by different interpolation methods. (a) Original image; (b) Bicubic (PSNR=30.14dB, SSIM=0.8550, FSIM=0.9651); (c) ScSR [6] (PSNR=30.00dB, SSIM=0.8472, FSIM=0.9609); (d) proposed NARM-SRM-NL (PSNR=31.16dB, SSIM=0.8693, FSIM=0.9699).

D. Discussions on parameter selection

In the proposed **Algorithm 1** for image interpolation, there are several parameters to be preset. In our implementation, the main parameters are set as: the patch size is set to 5×5 , the number of clusters is set to $K=60$, $\gamma=42000$, $\mu^{(0)}=1.4$, $\tau=1.2$, $\beta^{(0)}=0.1$, and $\rho=2$. We found that our algorithm is insensitive to these parameters in a reasonable range. Comparatively, the regularization parameters (i.e., λ and η) that balance the NARM based data fidelity term and the sparsity regularization terms are more critical to the performance of the proposed algorithm. In general, the larger the approximation error, the larger λ and η should be. In **Algorithm 1**, we initialize the iterative interpolation algorithm with $\lambda=0.1$ and $\eta=1.5$. After obtaining an initial estimate of the original image \mathbf{x} , denoted by $\hat{\mathbf{x}}$, we use $\hat{\mathbf{x}}$ to compute the adaptive regularization parameters λ_i and η_i using Eq. (13), where parameters c_1 and c_2 need to be preset. We found that the final interpolation result is insensitive to the initial regularization parameters λ and η , while setting c_1 and c_2 bigger will make the final results smoother and setting c_1 and c_2 smaller will make the convergence slow.

Figs. 17~18 show examples of the reconstructed HR images with different values of c_1 and c_2 . By experience, we set $c_1=0.25$ and $c_2=3.6$ to achieve a good balance between speed and good visual quality.

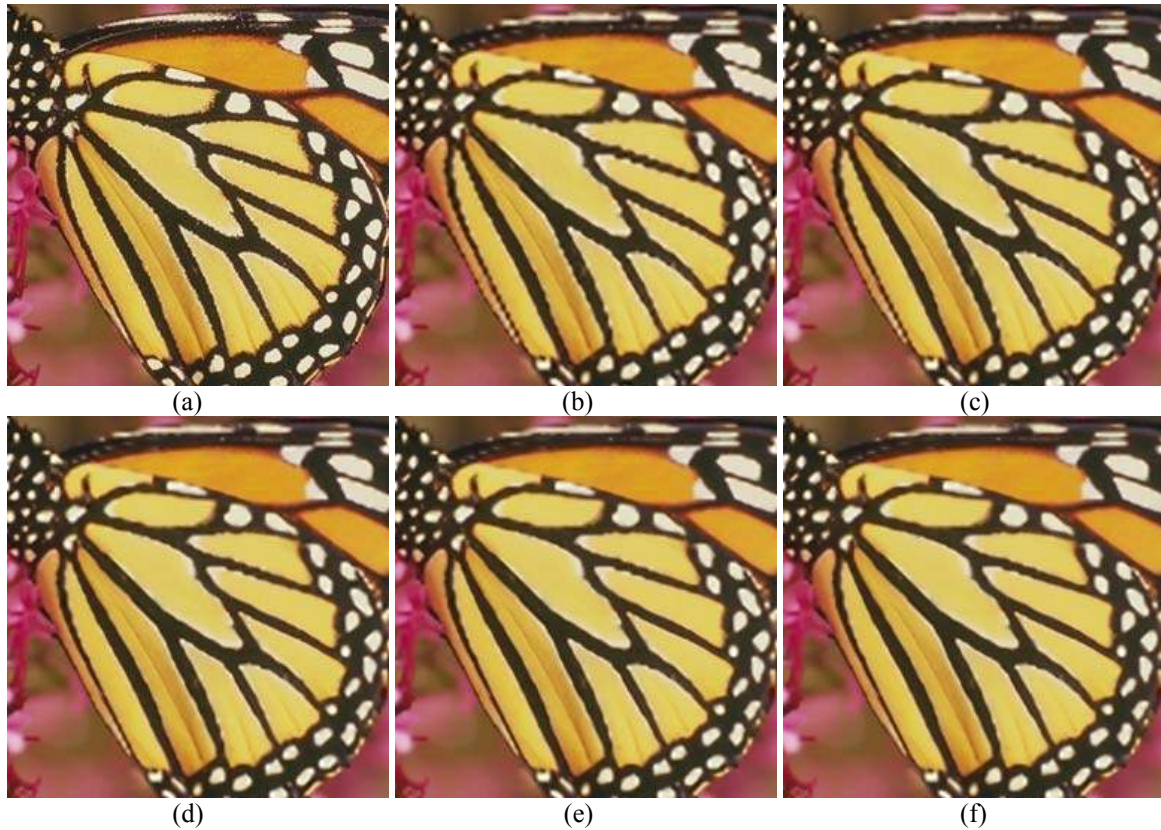


Figure 17: Effects of the regularization parameters. (a) Original image; the interpolated HR image (zooming factor 3) with the parameters: (b) $c_1=0.03$, $c_2=0.6$ (PSNR=24.93 dB); (c) $c_1=0.06$, $c_2=1.2$ (PSNR=25.16 dB); (d) $c_1=0.25$, $c_2=3.6$ (PSNR=25.45 dB); (e) $c_1=0.6$, $c_2=7.6$ (PSNR=25.56 dB); (f) $c_1=1.4$, $c_2=15.4$ (PSNR=25.57 dB).

6. Conclusion

In this paper we developed an effective image interpolation method by nonlocal autoregressive modeling (NARM) and embedding it in the sparse representation model (SRM). For the problem of image interpolation, the conventional SRM methods will become less effective because the data fidelity term will fail to impose structural constraint on the missing pixels. We addressed this issue by exploiting the image nonlocal self-similarity with NARM. By connecting a missing pixel with its nonlocal neighbors, the NARM can act as a new structural data fidelity term in SRM. We showed that NARM can reduce much the coherence between the sampling matrix and the sparse representation dictionary, making SRM more effective for image interpolation. Furthermore, we exploited the nonlocal redundancy to regularize the SRM

minimization, and used the local PCA dictionary to span adaptively the sparse domain for signal representation. Our extensive experimental results demonstrated that the proposed NARM method significantly outperforms state-of-the-art image interpolation methods in terms of both quantitative metrics and subjective visual quality.

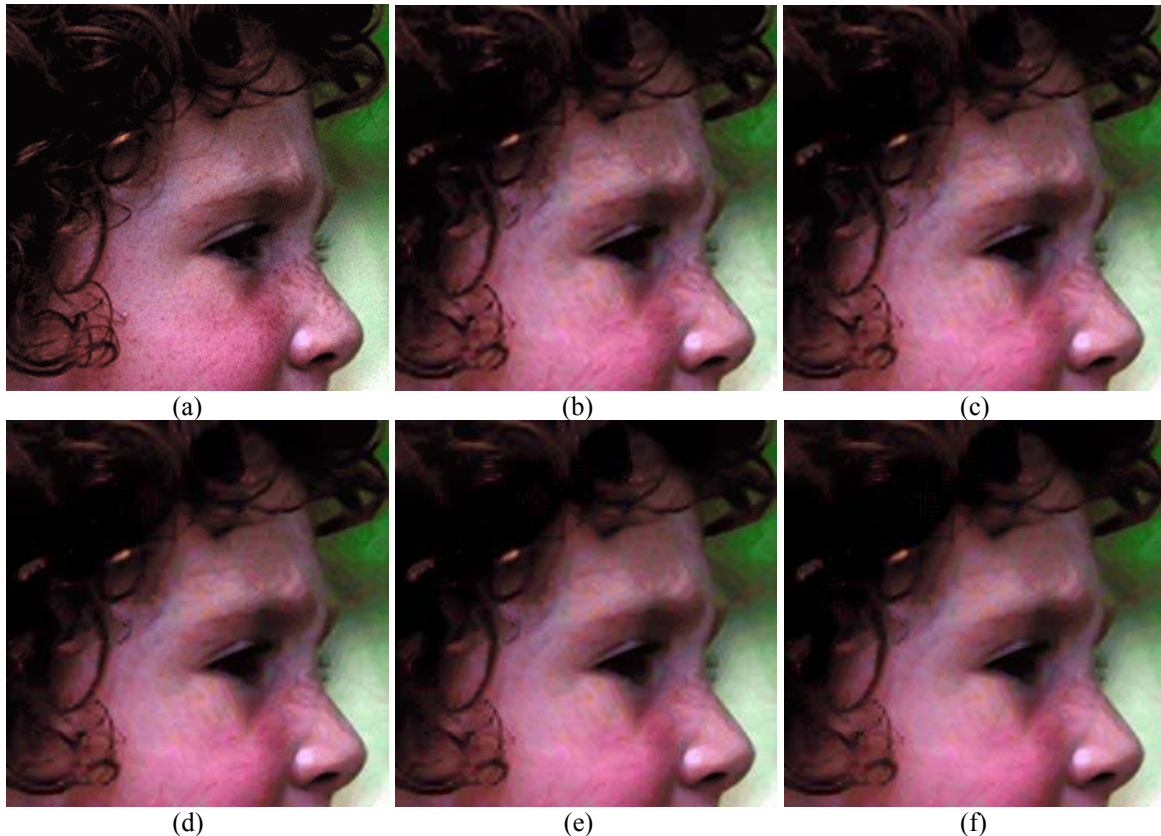


Figure 18: Effects of the regularization parameters. (a) Original image; the interpolated HR image (zooming factor 3) with the parameters: (b) $c_1=0.03$, $c_2=0.6$ (PSNR=31.93 dB); (c) $c_1=0.06$, $c_2=1.2$ (PSNR=31.87 dB); (d) $c_1=0.25$, $c_2=3.6$ (PSNR=31.64 dB); (e) $c_1=0.6$, $c_2=7.6$ (PSNR=31.05 dB); (f) $c_1=1.4$, $c_2=15.4$ (PSNR=29.98 dB).

Acknowledgement

This work is supported by the Hong Kong RGC General Research Fund (PolyU 5375/09E).

References

- [1] R. G. Keys, "Cubic convolution interpolation for digital image processing," *IEEE Trans. Acoustic, Speech, Signal Process.*, vol. ASSP-29, no. 6, pp. 1153-1160, Dec. 1981.
- [2] H. S. Hou and H. C. Andrews, "Cubic splines for image interpolation and digital filtering," *IEEE Trans. Acoustic, Speech, Signal Process.*, vol. 26, no. 6, pp. 508-517, Dec. 1978.
- [3] Xin Li and M. T. Orchard, "New edge-directed interpolation," *IEEE Trans. Image Processing*, vol. 10, pp. 1521-1527, Oct. 2001.

- [4] Lei Zhang and X. Wu, "An edge guided image interpolation algorithm via directional filtering and data fusion," *IEEE Trans. on Image Processing*, vol. 15, pp. 2226-2238, Aug. 2006.
- [5] X. Zhang and X. Wu, "Image interpolation by adaptive 2D autoregressive modeling and soft-decision estimation," *IEEE Trans. Image Process.*, vol. 17, no. 6, pp. 887-896, Jun. 2008.
- [6] Jianchao Yang, John Wright, Thomas Huang and Yi Ma, "Image Super-Resolution via Sparse Representation," *IEEE Trans. Image Process.* 2010.
- [7] W. Dong, L. Zhang, G. Shi, and X. Wu, "Image deblurring and super-resolution by adaptive sparse domain selection and adaptive regularization," *IEEE Trans. on Image Processing*, July 2011.
- [8] S. Mallat and G.Yu, "Super-Resolution with Sparse Mixing Estimators," *IEEE Trans. on Image Processing*, 2009.
- [9] M. Irani and S. Peleg, "Motion Analysis for Image Enhancement: Resolution, Occlusion, and Transparency," *Journal of Visual Communication and Image Representation*, vol. 4, no. 4, pp. 324-335, Dec. 1993.
- [10] L. Rudin, S. Osher, and E. Fatemi, "Nonlinear total variation based noise removal algorithms," *Phys. D*, vol. 60, pp. 259-268, 1992.
- [11] M. Lysaker and X. Tai, "Iterative image restoration combining total variation minimization and a second-order functional," *International Journal of Computer Vision*, vol. 66, no. 1, pp. 5-18, 2006.
- [12] A. Marquina, and S. J. Osher, "Image super-resolution by TV-regularization and Bregman iteration," *J. Sci. Comput.*, vol. 37, pp. 367-382, 2008.
- [13] A. Beck and M. Teboulle, "Fast gradient-based algorithms for constrained total variation image denoising and deblurring problems," *IEEE Trans. On Image Process.*, vol. 18, no. 11, pp. 2419-2434, Nov. 2009.
- [14] B. Olshausen and D. Field, "Emergence of simple-cell receptive field properties by learning a sparse code for natural images," *Nature*, vol. 381, pp. 607-609, 1996.
- [15] B. Olshausen and D. Field, "Sparse coding with an overcomplete basis set: a strategy employed by V1?" *Vision Research*, vol. 37, pp. 3311-3325, 1997.
- [16] I. Daubechies, M. Defriese, and C. DeMol, "An iterative thresholding algorithm for linear inverse problems with a sparsity constraint," *Commun. Pure Appl. Math.*, vol.57, pp.1413-1457, 2004.
- [17] P. Combettes, and V. Wajs, "Signal recovery by proximal forward-backward splitting," *SIAM J.Multiscale Model.Simul.*, vol.4, pp.1168-1200, 2005.
- [18] 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.
- [19] J. Mairal, M. Elad, and G. Sapiro, "Sparse Representation for Color Image Restoration," *IEEE Trans. on Image Processing*, vol. 17, no. 1, pages 53-69, Jan. 2008.
- [20] A. M. Bruckstein, D. L. Donoho, and M. Elad, "From sparse solutions of systems of equations to sparse modeling of signals and images," *SIAM Review*, vol. 51, no. 1, pp. 34-81, Feb. 2009.
- [21] E. Candès, M. B. Wakin, and S. P. Boyd, "Enhancing sparsity by reweighted l_1 minimization," *Journal of Fourier Analysis and Applications*, vol. 14, pp. 877-905, Dec. 2008.
- [22] M. Elad, M.A.T. Figueiredo, and Y. Ma, "On the Role of Sparse and Redundant Representations in Image Processing," *Proceedings of IEEE, Special Issue on Applications of Compressive Sensing & Sparse Representation*, June 2010.
- [23] R. Rubinstein, A.M. Bruckstein, and M. Elad, "Dictionaries for sparse representation modeling," *Proceedings of IEEE, Special Issue on Applications of Compressive Sensing & Sparse Representation*, vol. 98, no. 6, pp. 1045-1057, June, 2010.
- [24] E. Candès, J. Romberg, and T. Tao, "Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information," *IEEE Trans. on Information Theory*, vol. 52, no. 2, pp. 489 - 509, February 2006.
- [25] E. Candès and T. Tao, "Near optimal signal recovery from random projections: Universal encoding strategies?" *IEEE Trans. on Information Theory*, vol. 52, no. 12, pp. 5406 - 5425, December 2006.
- [26] J. Mairal, F. Bach, J. Ponce, G. Sapiro and A. Zisserman, "Non-Local Sparse Models for Image Restoration," *IEEE Int. Conference on Computer Vision*, Tokyo, Japan, 2009.
- [27] W. Dong, L. Zhang, and G. Shi, "Centralized sparse representation for image restoration," to appear in *IEEE Int. Conf. on Computer Vision (ICCV)*, 2011.
- [28] J. A. Tropp and S. J. Wright, "Computational methods for sparse solution of linear inverse problems," *Proceedings of IEEE, Special Issue on Applications of Compressive Sensing & Sparse*

- Representation*, vol. 98, no. 6, pp. 948-958, June, 2010.
- [29] S. Chen, D. Donoho, and M. Saunders, "Atomic decompositions by basis pursuit," *SIAM Review*, vol. 43, pp. 129-159, 2001.
- [30] 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.
- [31] A. Buades, B. Coll, J. M. Morel, "A non-local algorithm for image denoising," *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2005.
- [32] A. Buades, B. Coll, and J. M. Morel, "Nonlocal image and movie denoising," *Int. Journal Compute Vision (IJCV)*, vol. 76, pp. 123-139, 2008.
- [33] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, "Image denoising by sparse 3-D transform-domain collaborative filtering," *IEEE Trans. on Image Processing*, vol. 16, no. 8, pp. 2080-2095, Aug. 2007.
- [34] L. Zhang, W. Dong, D. Zhang, and G. Shi, "Two-stage image denoising by principal component analysis with local pixel grouping," *Pattern Recognition*, vol. 43, pp. 1531-1549, Apr. 2010.
- [35] S. Kindermann, S. Osher, and P. W. Jones, "Deblurring and denoising of images by nonlocal functionals," *Multiscale Modeling and Simulation*, vol. 4, no. 4, pp. 1091-1115, 2005.
- [36] X. Zhang, M. Burger, X. Bresson, and S. Osher, "Bregmanized nonlocal regularization for deconvolution and sparse reconstruction," 2009. *UCLA CAM Report (09-03)*.
- [37] M. Protter, M. Elad, H. Takeda, and P. Milanfar, "Generalizing the nonlocal-means to super-resolution reconstruction," *IEEE Trans. On Image Process.*, vol. 18, no. 1, pp. 36-51, Jan. 2009.
- [38] W. Dong, G. Shi, L. Zhang, and X. Wu, "Super-resolution with nonlocal regularized sparse representation," *SPIE Visual Communication and Image Processing*, vol. 7744, 2010.
- [39] R. Rubinstein, M. Zibulevsky, and M. Elad, "Double sparsity: Learning Sparse Dictionaries for Sparse Signal Approximation," *IEEE Trans. Signal Processing*, vol. 58, no. 3, pp. 1553-1564, March 2010.
- [40] N. Galatsanos and A. Katsaggelos, "Methods for choosing the regularization parameter and estimating the noise variance in image restoration and their relation," *IEEE Trans. on Image Process.*, vol. 1, no. 3, pp. 322-336, Mar. 1992.
- [41] R. Chartrand and W. Yin, "Iteratively reweighted algorithms for compressive sensing," *IEEE Int. Conf. on Acoustics, Speech and Signal Process. (ICASSP)*, pp. 3869-3872, Apr. 2008.
- [42] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: from error measurement to structural similarity," *IEEE Trans. Image Process.*, vol. 3, no. 4, pp. 600-612, Apr. 2004.
- [43] L. Zhang, L. Zhang, X. Mou, and D. Zhang, "FSIM: A Feature Similarity Index for Image Quality Assessment," *IEEE Trans. on Image Process.*, vol. 20, no. 8, pp. 2378-2386, Aug. 2011.
- [44] J. A. Tropp, "Algorithms for simultaneous sparse approximation," *Signal Processing*, vol. 86, pp. 572-602, 2006.
- [45] E. J. Candes and M. B. Wakin, "An introduction to compressive sampling," *IEEE Signal Processing Magazine*, pp. 21-30, Mar. 2008.
- [46] Y. Wang, J. Yang, W. Yin, and Y. Zhang, "A new alternating minimization algorithm for total variation image reconstruction," *SIAM J. Imaging Sciences*, vol. 1, no. 3, pp. 248-272, 2008.
- [47] D. P. Bertsekas, "Nonlinear programming," *Athena Scientific* 1999.
- [48] Z. Lin, M. Chen, L. Wu, and Y. Ma, "The augmented lagrange multiplier method for exact recovery of corrupted low-rank matrices," *UIUC Technical Report UILU-ENG-09-2215*, Oct. 2009.
- [49] D. Glasner, S. Bagon, and M. Irani, "Super-resolution from a single image," *IEEE Int. Conf. on Computer Vision*, pp. 349-356, Sep. 2009.
- [50] R. Zeyde, M. Elad, and M. Protter, "On single image scale-up using sparse representation," in *Curves & Surfaces*, Avignon-France, pp. 711-730, June 2010.