Image and Video Restorations via Nonlocal Kernel Regression

Haichao Zhang, *Student Member, IEEE*, Jianchao Yang, *Member, IEEE*, Yanning Zhang, *Senior Member, IEEE*, and Thomas S. Huang, *Life Fellow, IEEE*

***Abstract*—A nonlocal kernel regression (NL-KR) model is presented in this paper for various image and video restoration tasks. The proposed method exploits both the nonlocal self-similarity and local structural regularity properties in natural images. The nonlocal self-similarity is based on the observation that image patches tend to repeat themselves in natural images and videos, and the local structural regularity observes that image patches have regular structures where accurate estimation of pixel values via regression is possible. By unifying both properties explicitly, the proposed NL-KR framework is more robust in image estimation, and the algorithm is applicable to various image and video restoration tasks. In this paper, we apply the proposed model to image and video denoising, deblurring, and superresolution reconstruction. Extensive experimental results on both single images and realistic video sequences demonstrate that the proposed framework performs favorably with previous works both qualitatively and quantitatively.**

***Index Terms*—Deblurring, denoising, local structural regression, nonlocal self-similarity, restoration, superresolution (SR).**

I. Introduction

ONE OF the recent trends in image processing is to pursue the low-dimensional models for image representation and manipulation. Example works include the local-structure-based methods [1], [2], sparsity-based methods [3]–[11], and manifold-based methods [12], [13]. The success of these models is guaranteed by the low degree of freedom (DOF) of the local structures in natural images, represented as meaningful local structural regularity as well as nonlocal self-similarity of local patterns.

Many conventional image processing algorithms are based on the assumption of local structural regularity, which states that there are meaningful structures in the spatial space of natural images. Examples are bilateral filtering [14] and structure-tensor-based methods [1], [2], [15]–[17]. These methods utilize the local structural patterns to regularize the image processing procedure and are based on the assumption that images are locally smooth except at edges.

Along this direction, Tomasi proposed a bilateral filtering method for image filtering in [14], which exploits the local image structure during filtering. By augmenting the definition of the proximity between pixels by incorporating also the pixel values, rather than only the spatial locations, bilateral filtering overcomes the well-known blurring effect of a Gaussian filter and exhibits edge-preserving property, which is desirable for many image and video processing tasks. Tschumperlé [1] proposed a common framework for image restoration which is based on the iterative local diffusion in the image plane guided by the local structure tensor. Treating image restoration as a regression task on the 2-D image plane, Li and Orchard [15] and Takeda *et al.* [2] proposed respectively to improve the regression performance via regression kernels adapted to the local structures in the image. Li [16] further developed an implicit mixture motion model for video processing, which exploits the local spatial–temporal structures existing in videos. The generalization of 2-D kernel regression (KR) to three dimensions has also been studied in [17] for video superresolution (SR). To sum up, one common factor for the success of all these models is the exploration of the local image structures in images and videos.

Recently, another type of image processing methods exploiting the self-similarity in natural images is emerging. The self-similarity property means that higher level patterns, e.g., texton and pixon, will repeat themselves in the image. This also indicates that the DOF in the image is much lower than the DOF offered by the pixel-level representation. Such nonlocal self-similarity has been widely used in texture synthesis literatures [18], where the repetitive patterns are used to synthesize new texture regions. Recently, Buades and Coll have effectively applied this idea for image denoising, which is known as nonlocal means (NL-Means) method [19]. Different from the local KR method, NL-Means method breaks the locality constraint in the conventional restoration methods and estimates the pixel value from all the similar patches collected from a large region. It takes advantage of the redundancy of similar patches existing in the target image for the denoising task. More recently, this method has been generalized to handle multiframe SR task in [20]. This self-similarity property is also thoroughly explored by Glasner *et al.* in [21] for addressing single-image SR problems and general inverse problems in [22] by Peyré *et al.*

Manuscript received February 17, 2012; revised July 10, 2012 and September 22, 2012; accepted September 27, 2012. Date of publication November 12, 2012; date of current version May 10, 2013. This work was supported in part by the National Natural Science Foundation of China under Grants 61231016, 60872145, and 60903126, by the National High Technology Research and Development Program of China under Grant 2009AA01Z315, and by the U.S. Army Research Laboratory and Army Research Office under Grant W911NF-09-1-0383. The work of H. Zhang was supported by the Excellent Doctorate Foundation of Northwestern Polytechnical University. This paper was recommended by Associate Editor S. Zafeiriou.

H. Zhang and Y. Zhang are with the School of Computer Science, Northwestern Polytechnical University, Xi'an 710129, China (e-mail: hczhang@mail.nwpu.edu.cn; ynzhang@nwpu.edu.cn).

J. Yang is with the Advanced Technology Laboratory, Adobe Systems Inc., San Jose, CA 95110 USA (e-mail: jyang@ifp.uiuc.edu).

T. S. Huang is with the Beckman Institute, Department of Electrical and Computer Engineering, University of Illinois at Urbana–Champaign, Urbana, IL 61801 USA (e-mail: huang@ifp.uiuc.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TSMCB.2012.2222375

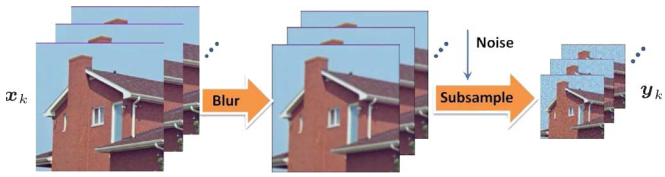

Fig. 1. Image formation process illustration. Typical image formation process includes the following: 1) Low-pass filtering; 2) subsampling; and 3) noise generation. The aim of restoration is to recover a high-quality image given the low-quality observation.

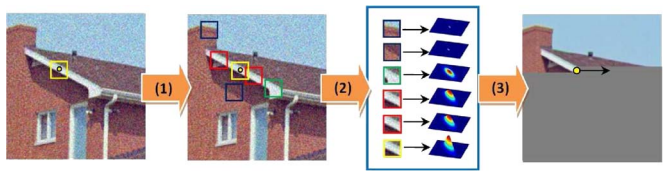

Fig. 2. Schematic illustration of NL-KR framework. (1) Similar patch searching: Different colors indicate the similarity (red denotes the highest, green the medium, and blue the least). (2) Structural kernel estimation and reweighting: Estimate a regression kernel adapted to the structure at each position where the similar patches reside and reweight them according to similarity. (3) NL-KR: Estimate the value for the query point with both local structural and nonlocal similar information in raster-scan order.

Image and video restorations are an important field in image and video processing community, which aims to estimate the high-quality version of the low-quality observations that are typically noisy and of low resolution (LR). Fig. 1 shows the typical image formation process assumed in the restoration literatures, where the low-quality observations are obtained by blurring and subsampling from the high-quality underlying image with sensing noise. This process can be modeled as follows:

$$\boldsymbol{y}_k = \boldsymbol{D}\boldsymbol{H}_k\boldsymbol{x}_k + \boldsymbol{\epsilon}_k = \boldsymbol{D}\boldsymbol{z}_k + \boldsymbol{\epsilon}_k, \qquad k = 1, 2, \ldots \quad (1)$$

where $\boldsymbol{x}_k$ represents the latent clean and sharp image while $\boldsymbol{y}_k$ denotes the corresponding noisy, blurry, and LR observation. $\boldsymbol{H}_k$ denotes the blurring operator corresponding to the convolutional blur kernel h_k, modeling the inherent low-pass filtering effects of the imaging system. $\boldsymbol{z}_k = \boldsymbol{H}_k\boldsymbol{x}_k$ denotes the blurry high-resolution (HR) image before subsampling. k is the frame/time-instance index. When $\boldsymbol{x}_k = \boldsymbol{x}\ \forall k = 1, 2, \ldots$, (1) models the formation process of multiple observations of the same static scene. $\boldsymbol{D}$ denotes the subsampling operator modeling the finite sampling property of practical imaging systems. $\boldsymbol{\epsilon}_k$ is usually assumed as an additive Gaussian noise term. In our preliminary work [23], we proposed a nonlocal KR (NL-KR) method for image and video SR. In this paper, we detail this to a complete general model which can be applied to different image and video restoration tasks, such as denoising, deblurring, and SR, with moderate adaptations. In our NL-KR model, we take advantage of both local structural regularity and nonlocal similarity in a unified framework for a more reliable and robust estimation. The nonlocal similarity and local structural regularity are intimately related and are also complementary in the sense that the fusion of the nonlocal similar patterns can be regularized by the structural regularity while the redundancy from similar patterns enables a more accurate estimation for structural regression. Fig. 2 shows a simplified schematic illustration of the proposed model.

The rest of this paper is organized as follows. We first review and summarize some related previous works for image and video restorations briefly in Section II; then, we explain our general NL-KR model in detail and discuss its relations to other algorithms in Section III. We detail the practical algorithm of NL-KR on several different restoration tasks in Section IV. Experiments are carried out in Section V on both synthetic and real image sequences, and extensive comparisons are made with both classical and *state-of-the-art* methods. Section VI provides some discussions and concludes this paper.

II. Image Restoration: Prior Art

In this paper, we are interested in image and video restorations where we desire to estimate the underlying latent high-quality image given the low-quality observation(s). Examples are image denoising, deblurring, and SR. This section presents a brief technical review of local structural regression or filtering method [1], [2], [15]–[17], [24] as well as the nonlocal-similarity-based approach [18]–[20], [25], [26], which are most closely related to our work.

The task of image restoration is to estimate the latent images $\{\boldsymbol{x}_k\}$ given the low-quality observations $\{\boldsymbol{y}_k\}$. For simplicity and clarity, in the following review of previous works, we take the single-observation-based denoising task as a canonical restoration problem without loss of generality. In the following, we use $\mathbf{x}_i \in \mathbb{N}^2$ to denote a general position in the 2-D image plane and italic bold lowercase letters (e.g., $\boldsymbol{x}$ and $\boldsymbol{y}$) to represent images. y_i is used as a shorthand for $\boldsymbol{y}(\mathbf{x}_i)$, i.e., the pixel value of $\boldsymbol{y}$ at $\mathbf{x}_i$.

A. Local Structural Regression

Typical image filtering methods usually perform in a local manner, i.e., the value of the estimated image at a query location is influenced only by the pixels within a small neighborhood of that position. They usually take the form of

$$\hat{\boldsymbol{z}}(\mathbf{x}_i) = \arg\min_{z_i} \sum_{j \in \mathcal{N}(\mathbf{x}_i)} (y_j - z_i)^2 K_{\mathbf{x}_i}(\mathbf{x}_j - \mathbf{x}_i) \quad (2)$$

where $\mathcal{N}(\mathbf{x}_i)$ denotes the neighbors of $\mathbf{x}_i$, y_j denotes the pixel observation at location $\mathbf{x}_j$, and $K_{\mathbf{x}_i}(\mathbf{x}_j - \mathbf{x}_i)$ is a generic spatial kernel at location $\mathbf{x}_i$ which typically assigns larger weights to nearby similar pixels while smaller weights to farther nonsimilar pixels. Regarding $\boldsymbol{z}(\mathbf{x}_i)$ as some regression function on the coordinate vector $\mathbf{x}_i$, (2) is essentially a zero-order estimation, where we directly minimize the weighted distance between $\boldsymbol{z}(\mathbf{x}_i)$ and y_j $(j \in \mathcal{N}(\mathbf{x}_i))$. In order to approximate the image local structure better, higher order estimation can be used [2], [24]. Specifically, assuming that the image is locally smooth to some order, we can rely on the local expansion of the function using the Taylor series [2]

$$\boldsymbol{z}(\mathbf{x}_i) = a_0 + \mathbf{a}_1^{\top}(\mathbf{x}_i - \mathbf{x}) + \mathbf{a}_2^{\top}\mathbf{tril}\left\{(\mathbf{x}_i - \mathbf{x})(\mathbf{x}_i - \mathbf{x})^{\top}\right\} + \cdots$$

where the regression coefficient a_0 is the function value (pixel value), $\mathbf{a}_1$ and $\mathbf{a}_2$ are defined as

$$\begin{aligned}\mathbf{a}_1 &= \nabla \boldsymbol{z}(\mathbf{x}) = \left[\frac{\partial \boldsymbol{z}(\mathbf{x})}{\partial x_1}, \frac{\partial \boldsymbol{z}(\mathbf{x})}{\partial x_2}\right]^\top \\ \mathbf{a}_2 &= \frac{1}{2}\left[\frac{\partial^2 \boldsymbol{z}(\mathbf{x})}{\partial x_1^2}, 2\frac{\partial^2 \boldsymbol{z}(\mathbf{x})}{\partial x_1 \partial x_2}, \frac{\partial^2 \boldsymbol{z}(\mathbf{x})}{\partial x_2^2}\right]^\top \end{aligned} \tag{3}$$

and $\mathbf{tril}(\cdot)$ denotes the operator that extracts the lower triangular part of a matrix and stacks it to a column vector. The optimal regression coefficients $\{\mathbf{a}_k\}_{k=0}^K$ can be estimated locally for effective approximation of images; thus, a spatial kernel should be applied, weighing the nearby samples larger than those farther away [2]

$$\begin{aligned}\{\mathbf{a}_k\}_{k=0}^K = \arg\min_{\{\mathbf{a}_k\}_{k=0}^K} \sum_{i=1}^m \Big[& y_i - a_0 - \mathbf{a}_1^\top(\mathbf{x}_i - \mathbf{x}) \\ & - \mathbf{a}_2^\top \mathbf{tril}\left\{(\mathbf{x}_i - \mathbf{x})(\mathbf{x}_i - \mathbf{x})^\top\right\} - \cdots \Big] K_{\mathbf{x}_i}(\mathbf{x}_i - \mathbf{x})\end{aligned}$$

where m denotes the number of pixels in the local neighborhood. It is easy to show that the aforementioned equation can be reformulated into a matrix form as a weighted least squares optimization problem

$$\begin{aligned}\hat{\mathbf{a}} &= \arg\min_{\mathbf{a}} E(\mathbf{a}) = \arg\min_{\mathbf{a}} \|R_{\mathbf{x}_i}\boldsymbol{y} - \Phi\mathbf{a}\|^2_{W_{K_{\mathbf{x}_i}}} \\ &= \arg\min_{\mathbf{a}} \left(R_{\mathbf{x}_i}\boldsymbol{y} - \Phi\mathbf{a}\right)^\top W_{K_{\mathbf{x}_i}} \left(R_{\mathbf{x}_i}\boldsymbol{y} - \Phi\mathbf{a}\right)\end{aligned} \tag{4}$$

where $R_{\mathbf{x}_i}$ is an operator which extracts the neighboring pixels centered at $\mathbf{x}_i$ (typically a patch) from $\boldsymbol{y}$ and represents it as a vector. $W_{K_{\mathbf{x}_i}}$ is the weight matrix induced from the kernel $W_{K_{\mathbf{x}_i}} = \mathrm{diag}[K_{\mathbf{x}_i}(\mathbf{x}_1 - \mathbf{x}_i), K_{\mathbf{x}_i}(\mathbf{x}_2 - \mathbf{x}_i), \ldots, K_{\mathbf{x}_i}(\mathbf{x}_m - \mathbf{x}_i)]$, with $m = |\mathcal{N}(\mathbf{x}_i)|$. $\mathrm{diag}[\cdot]$ constructs a diagonal matrix from the input vector, and Φ is the polynomial basis from the Taylor expansion [2]. It is also easy to see that the regression coefficient vector with respect to Φ is

$$\mathbf{a} = \left[a_0, \mathbf{a}_1^\top, \mathbf{a}_2^\top, \ldots\right]^\top. \tag{5}$$

To calculate the regression coefficient vector $\mathbf{a}$, we differentiate $E(\mathbf{a})$ with respect to $\mathbf{a}$

$$\frac{\partial E(\mathbf{a})}{\partial \mathbf{a}} = 2\Phi^\top W_{K_{\mathbf{x}_i}} \left(\Phi\mathbf{a} - R_{\mathbf{x}_i}\boldsymbol{y}\right) \tag{6}$$

and set it to be zero; then, we have

$$\hat{\mathbf{a}} = \left[\Phi^\top W_{K_{\mathbf{x}_i}} \Phi\right]^{-1} \Phi^\top W_{K_{\mathbf{x}_i}} R_{\mathbf{x}_i}\boldsymbol{y}. \tag{7}$$

According to the regression basis Φ and also (5), the first element of the regression coefficient vector $\hat{\mathbf{a}}$ is the desired pixel value estimation at $\mathbf{x}_i$; therefore

$$\begin{aligned}\hat{z}(\mathbf{x}_i) &= \hat{a}_0 = \mathbf{e}_1^\top \hat{\mathbf{a}} \\ &= \mathbf{e}_1^\top \left[\Phi^\top W_{K_{\mathbf{x}_i}} \Phi\right]^{-1} \Phi^\top W_{K_{\mathbf{x}_i}} R_{\mathbf{x}_i}\boldsymbol{y}\end{aligned} \tag{8}$$

where $\mathbf{e}_1$ is a vector with the first element equal to one and the rest is zero. This is the classical local structural regression, and it is a general formula which subsumes other approaches such as Gaussian filtering, bilateral filtering [14], and structure kernel guided filtering [1], [2], [24]. The review on local structural regression presented in this section mainly follows the work in [2], where more details on this topic can be found.

B. Nonlocal-Similarity-Based Estimation

Local image structures tend to repeat themselves within the image and also across the image sequence in videos, which has been explored in many applications such as texture synthesis [18], image inpainting, denoising [19], [25], and SR [20], [26]. Such self-similarity property provides the redundancy that is sometimes critical for many ill-posed image processing problems, as similar structures can be regarded as multiple observations from the same underlying ground truth signal. For instance, the NL-Means algorithm recently introduced by Buades and Coll in [19] for image denoising has become very popular, due to its effectiveness despite its simplicity. The algorithm breaks the locality constraints of previous conventional filtering methods, making use of similar patterns found in different locations of the image for denoising. Specifically, NL-Means estimates the pixel value of the current position as a weighted average of other similar pixels found by matching with not only their own pixel values but also their local neighboring pixels (i.e., patch matching)

$$\boldsymbol{z}(\mathbf{x}_i) = \frac{\sum_{j \in \mathcal{P}(\mathbf{x}_i)} w_{ij} y_j}{\sum_{j \in \mathcal{P}(\mathbf{x}_i)} w_{ij}} \tag{9}$$

where $\mathcal{P}(\mathbf{x}_i)$ denotes the index set for similar pixel observations for $\mathbf{x}_i$ found by patch matching (including $\mathbf{x}_i$ itself) and the weight w_{ij} reflects the similarity between the observations at $\mathbf{x}_i$ and $\mathbf{x}_j$ computed based on the similarity of the patches centered at $\mathbf{x}_i$ and $\mathbf{x}_j$, respectively [19]. Equation (9) can again be reformulated into a least squares optimization problem as follows [27]:

$$\begin{aligned}\hat{\boldsymbol{z}}(\mathbf{x}_i) &= \arg\min_{z_i} E(z_i) \\ &= \arg\min_{z_i} \sum_{j \in \mathcal{P}(\mathbf{x}_i)} \left[y_j - \boldsymbol{z}(\mathbf{x}_i)\right]^2 w_{ij} \\ &= \arg\min_{z_i} \|\mathbf{y} - \mathbf{1}\boldsymbol{z}(\mathbf{x}_i)\|^2_{W_{\mathbf{x}_i}} \\ &= \arg\min_{z_i} \left(\mathbf{y} - \mathbf{1}\boldsymbol{z}(\mathbf{x}_i)\right) W_{\mathbf{x}_i} \left(\mathbf{y} - \mathbf{1}\boldsymbol{z}(\mathbf{x}_i)\right)\end{aligned} \tag{10}$$

where $\mathbf{y}$ denotes the vector consisting of the pixels at the locations in the similar set $\mathcal{P}(\mathbf{x}_i)$, $\mathbf{1}$ denotes a vector of all ones, and $W_{\mathbf{x}_i} = \mathrm{diag}[w_{i1}, w_{i2}, \ldots, w_{im}]$, where $m = |\mathcal{P}(\mathbf{x}_i)|$. To show that (9) is the solution to (10), we differentiate $E(z_i)$ with respect to z_i

$$\frac{\partial E(z_i)}{\partial z_i} = 2 \times \mathbf{1}^\top W_{\mathbf{x}_i} \left(\mathbf{1}\boldsymbol{z}(\mathbf{x}_i) - \mathbf{y}\right). \tag{11}$$

Setting it to be zero and solving for z_i, we get

$$\boldsymbol{z}(\mathbf{x}_i) = \left[\mathbf{1}^\top W_{\mathbf{x}_i} \mathbf{1}\right]^{-1} \mathbf{1}^\top W_{\mathbf{x}_i} \mathbf{y}. \tag{12}$$

Since $W_{\mathbf{x}_i}$ is a diagonal matrix, thus (9) and (12) are equal.

Compared with (4), the NL-Means estimation (10) is a zero-order regression, since only the zero-order basis $\mathbf{1}$ is used for estimation. The associated weight matrix is constructed from the similarity measures over the similar patch set collected in a nonlocal fashion, instead of basing on the spatial kernel as in (4).

III. NL-KR Model

In this section, we will develop the proposed NL-KR method mathematically, followed with some discussions on its relations to some existing popular works.

A. Mathematical Formulation

In this section, we derive our NL-KR method, which makes full use of both cues from *local* structural regularity and *nonlocal* similarity for image and video restoration tasks. We state that the proposed approach is more reliable and robust for ill-posed inverse problems: Local structural regression regularizes the noisy candidates found by nonlocal similarity search, and nonlocal similarity provides the redundancy preventing possible overfitting of the local structural regression. Instead of using a *point prediction* model in the conventional nonlocal methods, we use the more reliable *local-structure-based* prediction. On the other hand, rather than predicting the value with only one *local patch*, we try to make use of all the *nonlocal similar* patches in natural images. Mathematically, the proposed high-order NL-KR model is formulated as

$$\begin{aligned}\hat{\mathbf{a}} &= \arg\min_{\mathbf{a}} E(\mathbf{a}) \\ &= \arg\min_{\mathbf{a}} \frac{1}{2}\overbrace{w_{ii}\left\|R_{\mathbf{x}_i}\boldsymbol{y}-\Phi\mathbf{a}\right\|^2_{W_{K_{\mathbf{x}_i}}}}^{\text{local}} \\ &\quad + \frac{1}{2}\overbrace{\sum_{j\in\mathcal{P}(\mathbf{x}_i)\setminus\{i\}} w_{ij}\left\|R_{\mathbf{x}_j}\boldsymbol{y}-\Phi\mathbf{a}\right\|^2_{W_{K_{\mathbf{x}_j}}}}^{\text{nonlocal}} \\ &= \arg\min_{\mathbf{a}} \frac{1}{2}\sum_{j\in\mathcal{P}(\mathbf{x}_i)} w_{ij}\left\|R_{\mathbf{x}_j}\boldsymbol{y}-\Phi\mathbf{a}\right\|^2_{W_{K_{\mathbf{x}_j}}} \\ &= \arg\min_{\mathbf{a}} \frac{1}{2}\sum_{j\in\mathcal{P}(\mathbf{x}_i)} \left\|R_{\mathbf{x}_j}\boldsymbol{y}-\Phi\mathbf{a}\right\|^2_{\tilde{W}_{\mathbf{x}_j}} \end{aligned} \tag{13}$$

where $W_{K_{\mathbf{x}_j}}$ is the weight matrix constructed from kernel $K_{\mathbf{x}_j}$ and $\mathcal{P}(\mathbf{x}_i)$ again is the index set for the patches similar to the patch centered at $\mathbf{x}_i$ (i.e., $R_{\mathbf{x}_i}\boldsymbol{y}$). w_{ij} is calculated between the location $\mathbf{x}_i$ of interest and another location $\mathbf{x}_j$ of a similar patch ($j \in \mathcal{P}(\mathbf{x}_i)$) by measuring the similarity of their neighborhoods weighted by a Gaussian kernel

$$w_{ij} = \exp\left(-\frac{\left\|R_{\mathbf{x}_i}\boldsymbol{y}-R_{\mathbf{x}_j}\boldsymbol{y}\right\|^2_{W_G}}{2\sigma^2}\right) \tag{14}$$

where W_G is the weight matrix constructed from a Gaussian kernel, which puts larger weights on the centering pixels of the patch. The proposed NL-KR regression model consists of two parts.

1) **Local regression term**: the traditional local regression or filtering term, with w_{ii} set to be one. This term also contributes as a fidelity loss, as the estimation should be close to the observation.
2) **Nonlocal regression term**: Instead of zero-order point estimation as in NL-Means, higher order KR is also used to make full use the structural redundancy in the similar patches.

To get the regression coefficients, we differentiate the energy function in (13) with respect to $\mathbf{a}$

$$\frac{\partial E(\mathbf{a})}{\partial \mathbf{a}} = \Phi^\top \sum_{j\in\mathcal{P}(\mathbf{x}_i)} w_{ij} W_{K_{\mathbf{x}_j}}\left(\Phi\mathbf{a} - R_{\mathbf{x}_j}\boldsymbol{y}\right). \tag{15}$$

Setting it to be zero, we can get the estimation for $\mathbf{a}$ as

$$\begin{aligned}\hat{\mathbf{a}} = &\left[\Phi^\top\left(\sum_{j\in\mathcal{P}(\mathbf{x}_i)} w_{ij}W_{K_{\mathbf{x}_j}}\right)\Phi\right]^{-1} \\ &\times \Phi^\top \sum_{j\in\mathcal{P}(\mathbf{x}_i)} w_{ij}W_{K_{\mathbf{x}_j}}R_{\mathbf{x}_j}\boldsymbol{y}.\end{aligned} \tag{16}$$

Then, $\hat{z}(\mathbf{x}_i) = \hat{a}_0 = \mathbf{e}_1^\top\hat{\mathbf{a}}$.

Examining (16), we have the following two comments.

1) The structural kernel is estimated from the contaminated observations and thus is not robust. Compared with (8), with nonlocal redundancy, our estimation is more stable because of the weighted average of kernel weight matrices inside the inverse and the weighted average of the structural pixel values.
2) Compared with (12), our model can regularize the estimation from the nonlocal patches by higher order structural regression and thus is more adaptive to the local context and more robust to outliers.

Therefore, the proposed model makes full use of both important cues from local structural regularity and nonlocal similarity, leading to more reliable and robust estimation, which will be verified by experimental results in Section V.

B. Structural Kernel Estimation

A natural design for the kernel would be an isotropic Gaussian kernel which assigns larger weights to spatially nearby pixels while giving smaller weights to the pixels far away. While achieving reasonable results on flat areas, such a kernel suffers from severe limitation that it usually blurs the edge structures in the image, as the support of the edge profile is usually smaller than that of the isotropic Gaussian kernel. Bilateral filtering [14] improves over isotropic Gaussian kernel by introducing the pixel value into the weight calculation other than the spatial coordinates, which shows to preserve the edge structures better. Moreover, it is desirable to make the kernel adaptive to the image structures, which can be achieved via the local gradient analysis [1], [2], [24]. Given the observation Y

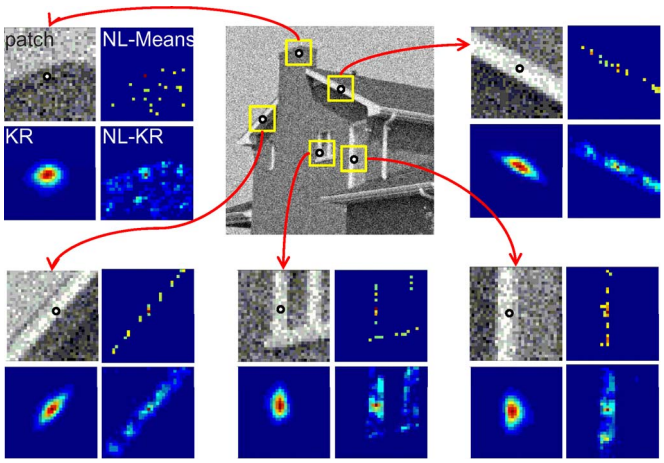

Fig. 3. Graphical illustration of the weight map computed from different methods. The black dot indicates the current position $\mathbf{x}_i$ under consideration. The patch size is 5×5. The yellow rectangle denotes the (nonlocal) spatial region for estimation. From the figure, we can see that the weight map for KR is localized with a single mode, where the peak is located at $\mathbf{x}_i$, and the orientation of the weight map is aligned with the underlying structure; the weight map from NL-Means scattered along the similar structure, without considering the local spatial support; and the weight map from NL-KR can have multiple modes and multiple disconnected components with spatial support.

and a query location $\mathbf{x}_i$, we can construct a structure adaptive kernel as

$$K_{\mathbf{x}_i}(\mathbf{x}-\mathbf{x}_i) = \frac{1}{2\pi s^2\sqrt{\det(T)}} \times \exp\left\{-\frac{1}{2s^2}(\mathbf{x}-\mathbf{x}_i)^\top T^{-1}(\mathbf{x}-\mathbf{x}_i)\right\}$$

where T is the diffusion tensor at $\mathbf{x}_i$ controlling the spatial structure of the kernel [1] and s is the parameter that controls the bandwidth of the structural kernel. Given two unit vectors $\mathbf{u}$ and $\mathbf{v}$ defined by the gradient and tangent directions, respectively, at the current position, we can construct the diffusion tensor as

$$T = f\mathbf{u}\mathbf{u}^\top + g\mathbf{v}\mathbf{v}^\top. \tag{17}$$

We can choose f and g according to the underlying local structure, so that the induced kernel is isotropic ($f \approx g$) at almost constant regions and aligned along the image contour ($g > f$) otherwise. One possible choice[1] is

$$f(\alpha,\beta) = \frac{\beta+\gamma}{\alpha+\gamma} \quad g(\alpha,\beta) = \frac{\alpha+\gamma}{\beta+\gamma} \qquad (\gamma \geq 0) \tag{18}$$

where α and β are the eigenvalues of the structure tensor [1], reflecting the strength of the gradient along each eigenvector direction. Specifically, α, β, $\mathbf{u}$, and $\mathbf{v}$ can be calculated from the following relation using eigenvalue decomposition:

$$\nabla \boldsymbol{y}_{\mathbf{x}_i} \nabla \boldsymbol{y}_{\mathbf{x}_i}^\top = \alpha\mathbf{u}\mathbf{u}^\top + \beta\mathbf{v}\mathbf{v}^\top \tag{19}$$

where $\nabla \boldsymbol{y}_{\mathbf{x}_i} \in \mathbb{R}^2$ is the gradient vector of $\boldsymbol{y}$ at $\mathbf{x}_i$.

A comparative illustration of the weight maps computed with NL-Means, KR, and NL-KR is shown in Fig. 3. From this figure, we can see that the weight map for KR is localized with a single mode, where the peak is located at $\mathbf{x}_i$, and the orientation of the weight map is aligned with the underlying structure; the weight map from NL-Means scattered at the similar structures, without considering the local spatial support; and the weight map from NL-KR could have multiple modes and multiple disconnected components with spatial support. From this illustration, we can see that the weight map from NL-KR assigns large weights to the pixels close to the current location $\mathbf{x}_i$, as well as those pixels close to the center of nonlocal similar patches, thus inducing a multiple-mode weight map.

[1]One can refer to [1] for other choices of diffusion tensor.

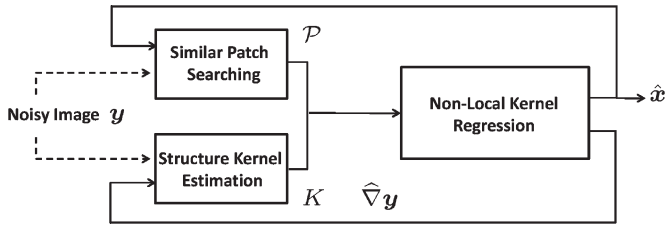

Fig. 4. Schematic illustration of the iterative NL-KR framework. Given a noisy input $\boldsymbol{y}$, the proposed method estimates the underlying noise-free data iteratively.

C. Iterative NL-KR

The structure kernel is data dependent, which manifests the nonlinear property of the proposed method, and consequently is sensitive to the noise in the input image. Similar to [2], we can iteratively estimate the structural kernel based on the refined estimation of the image from the previous iteration, as shown in Fig. 4. Although without an analysis of optimality and convergence, we experimentally validate such an iterative procedure on image denoising in Section V-A.

D. Relations to Other Works

Tons of works have emerged recently based on nonlocal redundancy and local structural regularity for image and video processing. It is worthwhile to talk about the relations of the proposed model to those previously proposed algorithms. The nonlocal models in [19], [20], and [28] use the redundancy from nonlocal self-similarity but do not include the spatial structure explicitly as a regularization. The high-order generalization of NL-Means in [27] uses the computation of nonlocal similarity to find the local kernel for regression, which actually violates the philosophy of the nonlocal model. Local-structure-based models [1], [2], [14], [24] explicitly employ the spatial kernel for regularization but neglect the redundancy of similar local patterns useful for robust estimation. The 3-D KR method [17] exploits the local spatial–temporal structure by extending their 2-D spatial KR model [2] and also discards the nonlocal self-similarity. Sparse representation for denoising [3] and SR [4] performs local regression using bases learned from a training database, which carries out estimation on each individual local patch and discards the patch redundancy. The sparse representation model is later generalized in [25] for image denoising by performing simultaneous sparse coding (SC) over similar patches found in different locations of the image. However, the

nonlocal redundancy is used in a hard assignment clustering way instead of a soft way. Glasner *et al.* [21] fully explore the self-similarity property for single-image SR, but no spatial structural regularization is applied. To summarize, our model is the first work to explicitly unify two important properties—the self-similarity and local structural regularity—into a single model, promoting more robust estimation and recovery.

IV. NL-KR for Image and Video Restorations

The NL-KR model proposed earlier is a general model that can be applied to different image and video restoration tasks. In this paper, we apply our model to denoising, deblurring, and SR tasks for images and videos.

A. Image Denoising and Deblurring

Image denoising is one of the most basic and fundamental problems in image processing. In this task, the observation process (1) is simplified as follows in the case of a single observation:

$$\boldsymbol{y} = \boldsymbol{x} + \boldsymbol{\epsilon} \tag{20}$$

where $\boldsymbol{\epsilon}$ is assumed as an additive Gaussian noise term. To denoise the noisy observation $\boldsymbol{y}$, a lot of different approaches have been developed in the literature, which mainly fall into one of the two categories mentioned in Section II. The first category is local filtering, with representative examples such as Gaussian filtering. To alleviate the inability of Gaussian filtering on structure preserving, local-structure-aware filtering methods are developed [1], [2], [14], [15]. The denoising methods by learning an overcomplete dictionary using image patches have shown to achieve promising results for denoising [29]. The second category includes the recent nonlocal denoising methods [19], [30]. These methods break the conventional locality constraint for estimation, which performs estimation based on similar patches collected in a nonlocal fashion.

For image denoising, our NL-KR model developed in Section III can be applied directly, by constructing the similar patch set $\mathcal{P}$ using the noisy observation $\boldsymbol{y}$, thus combining the local structural and nonlocal self-similarity information for robust estimation. Moreover, it is observed that the denoising result can be further improved by using the iterative NL-KR, i.e., constructing the similar patch set $\mathcal{P}$ using the denoising output of the last iteration and also updating the structural kernel estimation accordingly. This iterative method can generate denoising results comparable to *state-of-the-art* results, as demonstrated by the experimental results in Section V, although our method is not specifically designed for denoising task only. The main procedures for NL-KR denoising are presented in Algorithm 1 and also graphically shown in Fig. 4. The proposed method is mostly proper for handling Gaussian noise, due to our Gaussian assumption in the observation model. This is also true for the deblurring and SR scenario. Note that Algorithm 1 is only illustrated with images. However, it can be extended straightforwardly to video restoration task by constructing the similar patch set $\mathcal{P}$ spatially–temporally.

Algorithm 1 (NL-KR for image denoising).

1: **Input**: a noisy image $\boldsymbol{y}$, number of iterations N.
2: **Initialize** the current denoising estimation $\boldsymbol{x}^0 = \boldsymbol{y}$, estimate the image gradient $\nabla \boldsymbol{y}$ using $\boldsymbol{x}^0$;
3: **For** $t = 1, 2, \ldots, N$, do
4: **For** each pixel location $\mathbf{x}_i$ on the image grid, do
- Construct the similar patch index set $\mathcal{P}(\mathbf{x}_i)$ with current denoising estimation $\boldsymbol{x}^{t-1}$;
- Calculate the structure kernel $K_{\mathbf{x}_i}$ use (19) and (17);
- Construct the spatial weight matrix $\tilde{W}_{\mathbf{x}_j}$ using estimated $K_{\mathbf{x}_j}$ for all $j \in \mathcal{P}(\mathbf{x}_i)$;
- Calculate the regression coefficients with (16) and update the current estimation of $\boldsymbol{x}^t$ at $\mathbf{x}_i$ with $\boldsymbol{x}^t(\mathbf{x}_i) = \mathbf{e}_1^\top \hat{\mathbf{a}} = \hat{\mathbf{a}}(1)$;
- Update the image gradient $\nabla \boldsymbol{y}$ at $\mathbf{x}_i$ as $\nabla \boldsymbol{y}_{\mathbf{x}_i} = [\hat{\mathbf{a}}(2), \hat{\mathbf{a}}(3)]^\top$.

5: **End**
6: **End**
7: **Output**: a denoised image $\hat{\boldsymbol{x}} = \boldsymbol{x}^N$.

For the task of single-image deblurring, the observation process (1) can be simplified as follows:

$$\boldsymbol{y} = \boldsymbol{H}\boldsymbol{x} + \boldsymbol{\epsilon} = \boldsymbol{h} * \boldsymbol{x} + \boldsymbol{\epsilon} \tag{21}$$

where $\boldsymbol{\epsilon}$ is again an additive Gaussian noise term. Equation (21) can be rewritten equivalently in the frequency domain as

$$\mathcal{Y}(u,v) = \mathcal{H}(u,v)\mathcal{X}(u,v) + \mathcal{E}(u,v). \tag{22}$$

Assuming periodic boundary conditions, then $\boldsymbol{H}$ is a block circulant matrix with circulant blocks. The most direct way of estimating $\mathcal{X}$ is through inverse filtering

$$\hat{\mathcal{X}}(u,v) = \frac{\mathcal{Y}(u,v)}{\mathcal{H}(u,v)} = \mathcal{X}(u,v) + \frac{\mathcal{E}(u,v)}{\mathcal{H}(u,v)}. \tag{23}$$

However, due to the low-pass property of $\mathcal{H}$, many of its high-frequency elements are of small values or zeros. Therefore, taking its inverse as in (23) is very unstable and amplifies the high-frequency elements, thus suffering from severe artifacts known as "noise amplification." To attenuate the noise amplification, one can apply the Wiener filtering method for deblurring

$$\begin{aligned}\hat{\mathcal{X}}(u,v) &= \frac{\mathcal{H}^*(u,v)\mathcal{Y}(u,v)}{|\mathcal{H}(u,v)|^2 + \lambda} \\ &= \frac{|\mathcal{H}(u,v)|^2}{|\mathcal{H}(u,v)|^2 + \lambda}\mathcal{X}(u,v) + \frac{\mathcal{H}^*(u,v)}{|\mathcal{H}(u,v)|^2 + \lambda}\mathcal{E}(u,v)\end{aligned} \tag{24}$$

where $\mathcal{H}^*(u,v)$ denotes the complex conjugate of $\mathcal{H}(u,v)$. This is a simplified Wiener filtering using a flat signal spectrum with the value of λ. It is shown that, by choosing λ properly, one can recover most of the details for the image while not suffering much from severe noise amplification [31]. Following the combined Fourier-Wavelet Regularized Deconvolution (ForWaRD) method developed in [31], we apply our NL-KR

framework to image deblurring task as follows. Similar to the ForWaRD method, we first utilize Wiener filtering to get a rough deblurring estimation. Then, we estimate the final deblurring result from this estimation, where the main difference lies. Different from ForWaRD, which uses a wavelet shrinkage method for the final estimation, our proposed method takes advantage both the local and nonlocal information via NL-KR for the final estimation. The overall procedures are summarized in Algorithm 2. In practice, the regularization parameter λ is estimated via the same approach as used in ForWaRD [31]. Furthermore, iterative back-projection schemes as used in [5] and [32] can be used to further refine the deblurring results. As will be demonstrated by the deblurring results in Section V, the simple NL-KR deblurring method as proposed can achieve promising deblurring results without iterative back projection. Note that the proposed denoising and deblurring method can naturally handle videos by constructing the similar patch set via a spatial–temporal searching procedure.

Algorithm 2 (NL-KR for image deblurring)

1: **Input**: a blurry image $\boldsymbol{y}$, the blur kernel $\boldsymbol{h}$, regularization parameter λ.
2: **Initialize** frequency representation for $\boldsymbol{y}$ and $\boldsymbol{h}$ as $\mathcal{Y}$ and $\mathcal{H}$ using fast Fourier transformation (FFT).
3: **Perform** Wiener filtering using (24) to generate $\hat{\mathcal{X}}$;
4: **Generate** the spatial domain estimation $\tilde{\boldsymbol{x}}$ by applying inverse FFT to $\hat{\mathcal{X}}$;
5: **Perform** NL-KR denoising to $\tilde{\boldsymbol{x}}$ to generate the final deblurring result $\hat{\boldsymbol{x}}$;
6: **Output**: a deblurred image $\hat{\boldsymbol{x}}$.

B. SR

Image SR aims to estimate an HR image from a single or a set of LR observations and has many practical applications [33]. Conventional multiframe SR follows the steps of the following: 1) global motion estimation; 2) image wrapping; and 3) data fusion [34], [35]. These methods are limited in the assumed global motion model and cannot be applied to realistic videos that almost certainly contain arbitrary motion patterns. Recently, several multiframe SR algorithms based on fuzzy motion estimation of local image patches have been proposed to process real-world videos [17], [20]. We will show that, similarly, our model can also be applied to realistic videos, while achieving better results both qualitatively and quantitatively. Moreover, due to the self-similarity property of the image, we can also perform single-frame SR without additional training, arguing that the motion may not be that critical as in the conventional SR cases for image resolution enhancement. The LR image frames are usually modeled as blurring and downsampling the desired HR image, as modeled by (1). From this model, the SR recovery problem can be divided into two steps: LR image fusion and deblurring. In our NL-KR model, we also target recovering the blurry HR image $\boldsymbol{z}_k$ followed by deblurring. As now we have two different spatial scales, i.e., HR and LR image grids, the following notations are introduced for ease of presentation. We let r denote the zoom factor and $\mathbf{x}$ and $\underline{\mathbf{x}}$ denote the coordinates on HR and LR grids, respectively. R^h and R^l denote the patch extraction and vectorization operator on HR and LR images, where the extracted vectors are of dimension $u^2 \times 1$ and $v^2 \times 1$, respectively, and $u = (v-1) \times r + 1$ relates the two spatial scales. D_p is a patch downsampling operator which keeps the center pixel of the patch on the LR grid, while D_p^{T} is a patch upsampling operator with zero padding. Note that, when using matrix representation, D_p^{T} is the transpose of D_p. Please refer to Fig. 5 for the graphical illustrations of these operators. For a given query position $\mathbf{x}_i$ on the HR grid, $\mathcal{P}(\mathbf{x}_i)$ can be constructed from the initial HR estimation of the current image or consecutive frames, while keeping only those corresponding to integer positions on the LR grid, i.e., $\mathbf{x}_j = (\underline{\mathbf{x}}_j - 1) \times r + 1 (j \in \mathcal{P}(\mathbf{x}_i))$. Bicubic interpolation is used as initial HR estimation for similar patch set construction.

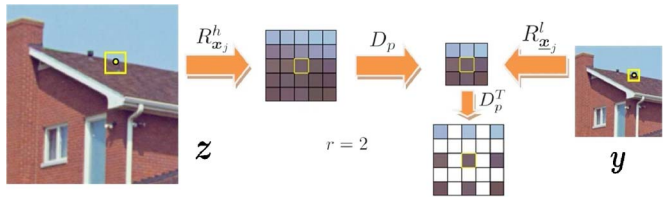

Fig. 5. Operator illustration. $R^l_{\underline{\mathbf{x}}_j}\boldsymbol{y}$, the patch of the LR image $\boldsymbol{y}$ at $\underline{\mathbf{x}}_j$, is formed by downsampling HR patch $R^h_{\mathbf{x}_j}\boldsymbol{z}$ by factor $r = 2$ as $D_p R^h_{\mathbf{x}_j}\boldsymbol{z}$, keeping the center pixel still in the center. Operator D_p^{T} up samples a patch with zero padding.

Using (1) and (13), the NL-KR model adapted for SR tasks is formulated as

$$\begin{aligned}\hat{\mathbf{a}} &= \arg\min_{\mathbf{a}} \frac{1}{2} \sum_{j \in \mathcal{P}(\mathbf{x}_i)} \left\| D_p^{\mathrm{T}} \left(R^l_{\underline{\mathbf{x}}_j}\boldsymbol{y} - D_p \Phi \mathbf{a} \right) \right\|^2_{\tilde{W}_{\mathbf{x}_j}} \\ &= \arg\min_{\mathbf{a}} \frac{1}{2} \sum_{j \in \mathcal{P}(\mathbf{x}_i)} \left(R^l_{\underline{\mathbf{x}}_j}\boldsymbol{y} - D_p \Phi \mathbf{a} \right)^{\mathrm{T}} \\ &\quad \times D_p \tilde{W}_{\mathbf{x}_j} D_p^{\mathrm{T}} \left(R^l_{\underline{\mathbf{x}}_j}\boldsymbol{y} - D_p \Phi \mathbf{a} \right) \\ &= \arg\min_{\mathbf{a}} \frac{1}{2} \sum_{j \in \mathcal{P}(\mathbf{x}_i)} \left\| R^l_{\underline{\mathbf{x}}_j}\boldsymbol{y} - D_p \Phi \mathbf{a} \right\|^2_{\tilde{W}^D_{\mathbf{x}_j}} \end{aligned} \tag{25}$$

where we denote $\tilde{W}_{\mathbf{x}_j} = w_{ij} W_{K_{\mathbf{x}_j}}$ to keep the notation uncluttered. $\Phi\mathbf{a}$ is a high-order regression for the patch $R^h_{\mathbf{x}_j}\boldsymbol{z}$ centered at query location $\mathbf{x}_j$ for the blurry HR image $\boldsymbol{z}$, and $\tilde{W}^D_{\mathbf{x}_j} = D_p \tilde{W}_{\mathbf{x}_j} D_p^{\mathrm{T}}$. Solution to (25) is

$$\begin{aligned}\hat{\mathbf{a}} = &\left[\Phi^{\mathrm{T}} \left(\sum_{j \in \mathcal{P}(\mathbf{x}_i)} D_p^{\mathrm{T}} \tilde{W}^D_{\mathbf{x}_j} D_p \right) \Phi \right]^{-1} \\ &\times \Phi^{\mathrm{T}} \sum_{j \in \mathcal{P}(\mathbf{x}_i)} D_p^{\mathrm{T}} \tilde{W}^D_{\mathbf{x}_j} R^l_{\underline{\mathbf{x}}_j}\boldsymbol{y}. \end{aligned} \tag{26}$$

The estimated pixel value at query point $\mathbf{x}_i$ is $\mathbf{e}_1^{\mathrm{T}}\hat{\mathbf{a}}$.

As we can see, the missing pixels on the HR grid are filled up by multiple LR observations found in a nonlocal way on the current frame or current sequence. These LR observations are further fused with regularization from the local structure.

The estimated image is then deblurred with a conventional deblurring algorithm to get the final estimation. We use total variation (TV)-based deblurring algorithm [36] in our experiment following [20] and [26] for fair comparison. Algorithm 3 describes the practical implementation for the proposed model.

Algorithm 3 (NL-KR for image SR)

1: **Input**: an LR video sequence $\underline{Y} = [\boldsymbol{y}_1, \boldsymbol{y}_2, \ldots, \boldsymbol{y}_k, \ldots]$, zoom factor r.
2: **Initialize** an enlarged sequence $\underline{\tilde{Y}} = [\tilde{\boldsymbol{y}}_1, \tilde{\boldsymbol{y}}_2, \ldots, \tilde{\boldsymbol{y}}_k, \ldots]$ with bicubic interpolation (BI) for all the LR frames.
3: **For** each pixel location $\mathbf{x}_i$ on the HR image grid for LR frame $\boldsymbol{y}_k$, do
   - Construct the similar patch index set $\mathcal{P}(\mathbf{x}_i)$ with sequence $\underline{\tilde{Y}}$;
   - Estimate the image gradient $\nabla\tilde{\boldsymbol{y}}_{\mathbf{x}_i}$;
   - Calculate the diffusion tensor $K_{\mathbf{x}_i}$ use (19) and (17).
4: **End**
5: **For** each pixel location $\mathbf{x}_i$ on the HR image grid for LR frame $\boldsymbol{y}_k$, do
   - Construct the spatial weight matrix $\tilde{W}^D_{\mathbf{x}_j}$ using estimated $K_{\mathbf{x}_j}$ for all $j \in \mathcal{P}(\mathbf{x}_i)$;
   - Calculate the regression coefficients with (26) and update the current estimation of $\boldsymbol{z}_k$ at $\mathbf{x}_i$ with $\boldsymbol{z}_k(\mathbf{x}_i) = \mathbf{e}_1^{\mathrm{T}}\hat{\mathbf{a}}$.
6: **End**
7: **Perform** deblurring for $\boldsymbol{z}_k$: $\boldsymbol{x}_k = \mathrm{TVdeblur}(\boldsymbol{z}_k)$.
8: **Output**: an HR video frame $\boldsymbol{x}_k$.

C. Discussions on the Implementation Details and Limitations

Implementation Details: We first detail some implementation issue in this section as well as the parameter settings. We set $\gamma = 1$ for diffusion tensor computation and the bandwidth of the structure kernel $s = 2.4$, which gives good performance empirically. A 5×5 patch is used in our implementation, and the number of similar patches is set as five. The size of search window should be set large enough to enable the nonlocal similarity to be exploited while not increasing the computational complexity too much. We found empirically that setting the search window size (radius) as three to four times the size of a patch can give desirable results. For denoising, we set the search window size as 20, while in SR, we use a search window size of 14. We empirically set $\sigma = 3$ for similarity weight calculation for denoising and $\sigma = 10$ for SR. For denoising, we set the iteration number as ten. The same number of iterations is used for image deblurring. For SR, we first use NL-KR for gradient estimation with isotropic structure kernel as initialization and then use the estimated gradients for SR estimation; therefore, it can be regarded as two NL-KR iterations. For video SR, we set the temporal search radius as four.

Limitations: As there is a patch matching procedure involved, the proposed method is not computationally efficient, which is a bottleneck shared by the methods using patch matching, e.g., [20]. Therefore, the current implementation is not very practical for real applications. However, this can be improved by using structured patch matching scheme proposed recently in [37], which explores the spatial structures in the matching correspondence for improving efficiency. Fast matching scheme such as K-dimensional (KD)-tree-based matching [38] can also be used. By using these speedup techniques, the computational complexity can be reduced from the current $O(nM^2)$ to about $O(nM \log M)$, making it more practical. Here, M denotes the size of the search region of the image, and n is the size of the patch. Another limitation is that, when applied to video restoration, the similar patches found temporally are treated the same way as the patches in the current frame. In other words, there is no *temporal-affinity* measure other than pure patch similarity measure embedded in our current model. By introducing a proper measure, the proposed model may fit the video applications better. Currently, our model is limited to spatial upscaling in terms of video SR. By incorporating a proper unified spatial–temporal similarity measure, temporal resolution enhancement may be achieved. Finally, the observation noise is assumed to be Gaussian in the current work. Proper similarity measure other than the Gaussian measure as used in the current work should be adopted for proper application of the proposed method in presence of non-Gaussian noise.

V. Experimental Validation

Experiments on image and video denoising, deblurring, and SR tasks are carried out to verify the effectiveness of the proposed method in this section.

A. Denoising Experiment

In this section, we apply the proposed NL-KR method to image denoising task. For denoising, we perform pixelwise value estimation using NL-KR on the 2-D image plane. We compare the performance of NL-KR with several *state-of-the-art* denoising methods. We first conduct a simulation experiment using standard test images. We generate a noisy image by adding white Gaussian noise with a standard deviation of 25 to the clean test image. Then, we perform denoising using the different algorithms mentioned previously. The results are summarized in Table I. Peak-signal-to-noise ratio (PSNR) is adopted as an evaluation metric for denoising results.[2] The Structural SIMilarity (SSIM) index [39] is also used for objective evaluation.

As can be seen from Table I, the proposed NL-KR method performs better than NL-Means [19] and KR [2] on denoising tasks, which use only the nonlocal and local information for estimation, respectively. This clearly verifies the effectiveness of exploiting both local structural regularity and the nonlocal self-similarity jointly as in the proposed method. The performance of the proposed NL-KR method is also better than that of the K-means singular value decomposition method [29], which is a learning-based denoising method and requires additional training phase before denoising. Moreover, the denoising result

[2]PSNR is defined as $\mathrm{PSNR} = 10\log_{10}(255^2/(1/M)\sum_i(\hat{\boldsymbol{x}} - \boldsymbol{x})_i^2)$, where $\hat{\boldsymbol{x}}$ denotes the denoised estimation and $\boldsymbol{x}$ is the original clean image. M is the total number of pixels in $\boldsymbol{x}$.

TABLE I
DENOISING RESULTS MEASURED IN TERMS OF (TOP) PSNR AND (BOTTOM) SSIM

Image	NL-Means	KR	KSVD	BM3D	Proposed
Lena	30.33	31.56	31.34	32.04	31.87
Barbara	28.48	28.53	29.54	30.64	29.69
Boat	28.50	29.12	29.21	29.84	29.52
Pepper	29.07	29.27	29.54	30.23	29.94
House	30.62	31.49	32.12	32.92	32.43
Cameraman	28.26	27.83	28.52	29.11	28.79
Lena	0.7712	0.8522	0.8432	0.8595	0.8572
Barbara	0.7952	0.8455	0.8499	0.8859	0.8582
Boat	0.7284	0.7824	0.7712	0.8005	0.7886
Pepper	0.7994	0.8479	0.8536	0.8642	0.8575
House	0.7678	0.8317	0.8445	0.8563	0.8497
Cameraman	0.7709	0.8046	0.8260	0.8454	0.8331
Time (s)	63.6	11.1	114.1	1.1	572.8

of our method is comparable to block-matching and 3-D filtering (BM3D) [40] in terms of PSNR and SSIM, which is one of the *state-of-the-art* denoising algorithms in the literature. It is worth mentioning that BM3D has an additional step of Wiener filtering after its 3-D collaborative filtering step to boost its denoising performance, while our proposed method without such an additional step can already generate results comparable to the results of BM3D. The average computational time per iteration for each algorithm over the 256×256-sized images on a desktop PC with Intel Core2 Duo CPU E6500 (2.33 GHz) and 4-GB RAM is also reported in Table I for reference. Our currently unoptimized implementation has a relatively high computational cost, as also mentioned when discussing the limitations of the proposed method in Section IV-C. Future directions for improving this have been discussed in Section VI.

With image denoising as a test bed, we demonstrate the effectiveness of combining local and nonlocal information jointly with the proposed NL-KR method compared with two other two-step configurations: 1) KR followed by NL-Means and 2) NL-Means followed by KR. These two configurations can also exploit the local and nonlocal information but in a sequential way rather than a joint way as in our approach. The results are shown in Fig. 6. As KR requires multiple iterations, we select among 15 iterations for both 1) and 2) the best result in terms of the final output after the two separate sequential steps. It is demonstrated that, with a proper number of iterations (which is typically smaller than that required by pure KR), both of the two-step approaches can improve over the NL-Means and KR, respectively. However, the joint estimation with the proposed method is more effective for restoration, outperforming both of the two-step configurations.

B. Deblurring Experiment

In this section, we carry out several experiments to evaluate the efficacy of the proposed method for image deblurring. Increase in signal-to-noise ratio (ISNR) is used as the objective metric to evaluate the deblurring results, which is widely used in image deblurring problem.[3] The experimental setups are as follows. A 9×9 box blur is used for generating the blurry

Fig. 6. Benefits of joint estimation with NL-KR compared with the separate two-step approaches. Left to right: (a) KR followed by NL-Means (31.62, 0.8380), (b) NL-Means followed by KR (31.52, 0.8374), and (c) joint estimation with NL-KR (32.43, 0.8497). The joint estimation with NL-KR can recover the noise-free image better than the other two-step separate estimation schemes.

TABLE II
DEBLURRING RESULTS IN TERMS OF ISNR

BSNR	Image	Wiener	ForWaRD	BM3D	Proposed
20 dB	Peppers	7.36	8.14	8.19	8.15
	House	7.28	8.82	9.02	8.82
	Cameraman	5.40	6.33	6.43	6.23
30 dB	Peppers	9.55	11.27	11.54	11.18
	House	8.77	11.76	12.53	12.24
	Cameraman	7.23	8.50	8.80	8.86
40 dB	Peppers	11.02	13.93	14.20	14.11
	House	9.04	13.82	14.54	14.36
	Cameraman	9.00	10.77	11.01	10.97

Fig. 7. Image deblurring (ISNR, SSIM for "cameraman" image in brackets). Left to right: (a) Blurry and noisy images ($\mathrm{BSNR} = 40$ dB), (b) ForWaRD [31] (10.77, 0.8346), (c) BM3D (11.01, 0.8489), (d) BM3D-DB [41] (11.79, 0.8647), and (e) NL-KR (10.97, 0.8369).

image. Then, Gaussian noise is added to the blurry image with blurred signal-to-noise ratio (BSNR) as {20, 30, 40} dB.[4] We compare the proposed method with other methods: classical Wiener filtering, ForWaRD [31], and BM3D [40] methods.[5]

The results are summarized in Table II. As can be seen from the results in Table II, the proposed method performs similar to ForWaRD method when the noise level is high. With lower noise level, the proposed method performs better than the ForWaRD method and compares favorably to the BM3D method. We also show some of the deblurred results in Fig. 7, where we have also included the deblurring results using the method in [41]. As can be seen from this figure, the ForWaRD method can remove most of the amplified noise via a further shrinkage step in wavelet domain. However, this method suffers from some blocky effects along the edge structures, due to the point singularity property of wavelet transformation. Due to the incorporation of both local structural and nonlocal similarity information, the proposed NL-KR method can avoid the noise amplification during deblurring while can retain the detail

[3]ISNR quantifies the SNR difference between the deblurred image $\hat{\boldsymbol{x}}$ and the blurry image $\boldsymbol{y}$: $\mathrm{ISNR} = 10\log_{10}(\sum_i (\boldsymbol{y}-\boldsymbol{x})_i^2 / \sum_i (\hat{\boldsymbol{x}}-\boldsymbol{x})_i^2)$.

[4]BSNR, defined as $\mathrm{BSNR} = 10\log_{10}(\sum_i \boldsymbol{\epsilon}_i^2 / \sum_i (\hat{\mathbf{x}}-\boldsymbol{x})_i^2)$.

[5]For BM3D method, we keep all the other settings the same as NL-KR method while just replacing the filtering scheme in Algorithm 2 with BM3D.

TABLE III
(TOP) PSNR AND (BOTTOM) SSIM RESULTS FOR SINGLE-IMAGE SR

Image	NN	BI	GNLM [20]	KR [2]	SC [4]	Proposed
Foreman	28.69	30.95	32.00	32.45	32.60	32.76
Miss A.	31.58	34.07	34.47	34.44	34.91	35.40
Suzie	30.03	31.47	31.65	31.82	31.52	32.10
Foreman	0.8185	0.8708	0.8747	0.8862	0.8768	0.8924
Miss A.	0.8403	0.8941	0.9008	0.8990	0.8843	0.9117
Suzie	0.7892	0.8286	0.8355	0.8285	0.8334	0.8449

structures well. However, as we have also noticed from Fig. 7, the performance of our method is not as good as the method in [41]. The reason is that our current NL-KR model is suitable for handling Gaussian noise, while the noise after the Wiener filtering is no longer Gaussian but colored. Taking this fact into consideration may further improve the deblurring result.

C. SR Experiment

The proposed NL-KR model can handle both single-image and multiple-frame SRs naturally. In this section, we validate the performance of the proposed method with experiments on single images and synthetic and real-world video sequences. Performance comparisons are performed with related *state-of-the-art* algorithms. We use both PSNR and SSIM index [39] to evaluate different algorithms objectively.

In all the experiments, we focus on zooming the LR frame(s) by a factor of three. These LR frames are modeled by first blurring the HR frames with a 3×3 uniform blur kernel and downsampling with a decimation factor of three. Gaussian noise with a standard deviation of two is added to the LR frames to model the real-world imaging system. In our algorithm, the LR patch size is fixed as 5×5, and the corresponding HR patch size is thus 13×13. The support of the nonlocal similar patch searching is fixed to be the ten nearest neighbors (NNs). We set $\sigma = 169c$ with $c = 0.06$ for similarity weight calculation. For image deblurring, we use a TV-based deblurring algorithm [36].[6]

1) Single-Frame-Based SR: We first evaluate the proposed method on single-image SR task. In the first set of experiments, we specifically compare the proposed model with the 2-D case of Generalized NL-Means (GNL-Means) [20] and KR [2] in order to show that our model is more reliable and robust for estimation. We take the first frame from each of the test sequences: Foreman, Miss America, and Suzie used in [20], degrade it, and perform the SR estimation. The PSNR and SSIM results for the three frames are summarized in Table III, which shows that the proposed method performs constantly better than 2-D GNL-Means and 2-D KR. The results of NN interpolation, BI, and SC methods [4] are also provided as references. Fig. 8 shows the visual quality comparisons on Foreman. As shown, the 2-D GNL-Means method is prone to block artifacts due to poor patch matching within a single image, and the 2-D KR method generates ghost effects due to insufficient observation for regularizing the local regression. Our result, however, is free of either of these artifacts.

[6]Note that, while the deblurring approach proposed in Section IV-A can be applied, TV-based deblurring is used here following [20] for fair comparison.

Fig. 8. Single-frame SR (×3, PSNR, SSIM in brackets). Left to right: (a) NN (28.69, 0.8185), (b) BI (30.95, 0.8708), (c) GNL-Means (32.00, 0.8747) [20], (d) KR (32.45, 0.8862) [2], (e) SC (32.60, 0.8768) [4], and (f) NL-KR (32.76, 0.8918). GNL-Means generates block effect, while KR generates ghost artifacts. Our method does not suffer from these problems.

Fig. 9. Single-frame SR for real color images (×4). Left to right: (a) NN, (b) BI, (c) Kim's method [42], (d) Fattal's method [43], (e) Glasner's method [21], and (f) NL-KR. Note that our result preserves more details than Fattal's method and is comparable to the results from Kim's learning-based method and recently proposed method by Glasner.

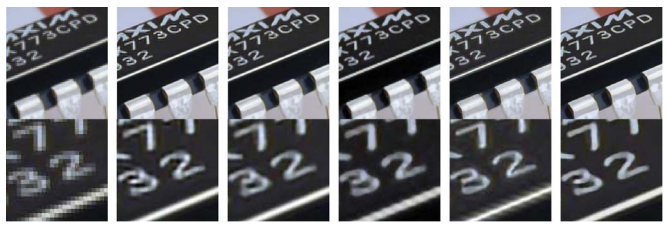

Fig. 10. More results on single-frame SR for real color images (×4). Left to right: (a) NN, (b) GNL-Means [20], (c) KR [2], (d) Fattal's method [43], (e) Glasner's method [21], and (f) NL-KR. Note that Fattal's method and Glasner's method generate *jaggy* effects on the edge. Our method is free from the *jaggy* artifacts and preserves better structure.

We further make more comparisons with *state-of-the-art* methods on real-world images, where the input LR image is zoomed by a factor of four, as shown in Figs. 9 and 10. Note that these methods are designed specifically to work on single images. In Fig. 9, it can be seen that the proposed method can preserve more details than Fattal's method [43] and is comparable with Kim's method [42] and the more recent work [21]. In Fig. 10, however, our algorithm outperforms both those in [43] and [21], where our result is free of the *jaggy* artifacts on the edges and the characters generated by our method are more realistic. The improvement comparison could be more impressive if one notices that, in [21], multiple scales are used for similar pattern matching while our method only uses one scale, although our results can be further improved by using multiscale self-similarities [44].

2) Synthetic Experiment for Multiframe SR: Our second experiment is conducted on synthetic image frames. We generate nine LR images from one HR image by blurring the HR image with a 3×3 uniform blur kernel and then decimate the blurred HR image every third row or column with shifts of {0, 1, 2} pixels. Gaussian noise with a standard deviation of two is

TABLE IV
(TOP) PSNR AND (BOTTOM) SSIM RESULTS FOR SYNTHETIC TEST FRAMES

Sequence	NN	BI	GNLM [20]	BM3D [30]	Proposed
Foreman	28.90	30.95	34.68	34.90	35.20
Miss A.	31.60	34.07	36.25	37.50	37.82
Suzie	30.03	31.47	32.92	33.60	33.99
Foreman	0.8413	0.8709	0.9044	–	0.9234
Miss A.	0.8404	0.8928	0.8193	–	0.9346
Suzie	0.7904	0.8290	0.8428	–	0.8864

TABLE V
AVERAGE (TOP) PSNR AND (BOTTOM) SSIM FOR THE THREE VIDEO SEQUENCES

Sequence	NN	BI	GNLM [20]	BM3D[30]	Proposed
Foreman	28.84	31.05	32.82	33.50	34.01
Miss A.	31.66	34.24	35.35	36.30	36.44
Suzie	30.08	31.44	32.97	33.00	33.09
Foreman	0.8207	0.8720	0.9025	–	0.9120
Miss A.	0.8426	0.8938	0.9136	–	0.9164
Suzie	0.7857	0.8233	0.8797	–	0.8671

also added. The PSNR and SSIM results are summarized in Table IV, showing that the proposed method is again constantly better. Note that the results from BM3D are cited directly from [30], which are obtained from noise-free observations.

3) Evaluation on Real-World Video Sequences: Finally, we evaluate the performance of our model on three real image sequences with general motions: Foreman, Miss America, and Suzie. Comparisons are made with the GNL-Means [20], BM3D [30], and 3D-KR [17]. The average PSNR and SSIM results on these three test sequences are reported in Table V. As shown, the proposed method achieves better reconstruction accuracy than GNL-Means and BM3D. The SR results on Foreman sequences are shown in Fig. 11 for visual comparison. Note that GNL-Means sometimes generates severe block artifacts (see the *Mouth* part in Fig. 11). The 3D-KR method, on the other hand, will generate some *ghost* effects, due to overfitting of the regression and inaccurate estimation of the 3-D kernel (see the *Mouth* part in Fig. 11). Furthermore, the 3D-KR method has to employ a motion precompensation procedure in order for good 3-D kernel estimation, while our model does not require such a step. Finally, the BM3D method generates severe artifacts at edge areas, while our method can recover the edge structures much better. Our method is also better than BM3D in terms of objective evaluation. This clearly shows the advantage of our method over BM3D on SR task.

VI. DISCUSSIONS AND CONCLUSION

This paper has proposed an NL-KR model for image and video restoration tasks, which combines the local structural regularity and nonlocal similarity explicitly to ensure a more reliable and robust estimation. The proposed method is a general model that includes many related models as special cases. We apply the proposed method to image and video denoising, deblurring, and SR tasks in this work. Extensive experimental results compared with *state-of-the-art* methods for each task demonstrate the generality and effectiveness of our model. In the current algorithm, the patch matching and spatial kernel calculation are most computationally heavy, which can be speeded up by KD-tree [38] searching and parallel computing, respectively. Furthermore, we can improve this by using the structured patch matching scheme proposed in [37] and [45] and by exploring the spatial structures in the matching correspondence. Our future work would include developing fast algorithm of the proposed method and applying the proposed method to other related applications.

Fig. 11. Video SR for Foreman sequence (frame 8, zoom factor of three, PSNR and SSIM in brackets). Left to right: (a) Ground truth, (b) BI (30.18, 0.8739), (c) GNL-Means (33.22, 0.9041), (d) BM3D (33.45, NA) (cited from [17]), (e) 3D-KR (33.30, NA), and (f) NL-KR (33.76, 0.9137). GNL-Means performs well at the regularly structured area but generates severe block artifacts where few similar patches can be found, BM3D suffers from *jagged* effects at the edges, 3D-KR cannot preserve the straight structure well due to the nonrobustness of its spatial–temporal kernel and can generate *ghost image*, and our method preserves both the larger structure and fine details well and is free of these artifacts.

REFERENCES

[1] D. Tschumperlé, "PDE's Based Regularization of Multivalued Images and Applications," Ph.D. Thesis, Université de Nice Sophia-Antipolis, Nice, France, 2002.

[2] H. Takeda, S. Farsiu, and P. Milanfar, "Kernel regression for image processing and reconstruction," *IEEE Trans. Image Process.*, vol. 16, no. 2, pp. 349–366, Feb. 2007.

[3] M. Elad and M. Aharon, "Image denoising via learned dictionaries and sparse representation," in *Proc. IEEE Conf. CVPR*, 2006, pp. 895–900.

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

[5] H. Zhang and Y. Zhang, "Sparse representation based iterative incremental image deblurring," in *Proc. IEEE ICIP*, 2009, pp. 1293–1296.

[6] V. M. Patel, G. R. Easley, and D. M. Healy, Jr., "Shearlet-based deconvolution," *IEEE Trans. Image Process.*, vol. 18, no. 12, pp. 2673–2685, Dec. 2009.

[7] H. Zhang, J. Yang, Y. Zhang, and T. S. Huang, "Sparse representation based blind image deblurring," in *Proc. IEEE ICME*, 2011, pp. 1–6.

[8] H. Zhang, J. Yang, Y. Zhang, N. M. Nasrabadi, and T. S. Huang, "Close the loop: Joint blind image restoration and recognition with sparse representation prior," in *Proc. IEEE ICCV*, 2011, pp. 770–777.

[9] H. Zhang, Y. Zhang, and T. S. Huang, "Exploiting structured sparsity for image deblurring," in *Proc. IEEE ICME*, 2012, pp. 616–621.

[10] J. Yang, Z. Wang, Z. Lin, S. Cohen, and T. S. Huang, "Coupled dictionary training for image super-resolution," *IEEE Trans. Image Process.*, vol. 21, no. 8, pp. 3467–3478, Aug. 2012.

[11] H. Zhang, Y. Zhang, H. Li, and T. S. Huang, "Generative Bayesian image super resolution with natural image prior," *IEEE Trans. Image Process.*, vol. 21, no. 9, pp. 4054–4067, Sep. 2012.

[12] G. Peyré, "Manifold models for signals and images," *Comput. Vis. Image Und.*, vol. 113, no. 2, pp. 249–260, Feb. 2009.

[13] J. Ni, P. K. Turaga, V. M. Patel, and R. Chellappa, "Example-driven manifold priors for image deconvolution," *IEEE Trans. Image Process.*, vol. 20, no. 11, pp. 3086–3096, Nov. 2011.

[14] C. Tomasi, "Bilateral filtering for gray and color images," in *Proc. ICCV*, 1998, pp. 839–846.

[15] X. Li and M. Orchard, "New edge-directed interpolation," *IEEE Trans. Image Process.*, vol. 10, no. 10, pp. 1521–1527, Oct. 2001.

[16] X. Li, "Video processing via implicit and mixture motion models," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 17, no. 8, pp. 953–963, Aug. 2007.

[17] H. Takeda, P. Milanfar, M. Protter, and M. Elad, "Super-resolution without explicit subpixel motion estimation," *IEEE Trans. Image Process.*, vol. 18, no. 9, pp. 1958–1975, Sep. 2009.

[18] A. Efros and T. Leung, "Texture synthesis by non-parametric sampling," in *Proc. IEEE ICCV*, 1999, pp. 1033–1038.
[19] A. Buades and B. Coll, "A non-local algorithm for image denoising," in *Proc. IEEE CVPR*, 2005, pp. 60–65.
[20] M. Protter, M. Elad, H. Takeda, and P. Milanfar, "Generalizing the non-local-means to super-resolution reconstruction," *IEEE Trans. Image Process.*, vol. 18, no. 1, pp. 36–51, Jan. 2009.
[21] D. Glasner, S. Bagon, and M. Irani, "Super-resolution from a single image," in *Proc. IEEE ICCV*, 2009, pp. 349–356.
[22] G. Peyré, S. Bougleux, and L. Cohen, "Non-local regularization of inverse problems," in *Proc. ECCV*, 2008, pp. 57–68.
[23] H. Zhang, J. Yang, Y. Zhang, and T. S. Huang, "Non-local kernel regression for image and video restoration," in *Proc. ECCV*, 2010, pp. 566–579.
[24] T. Q. Pham, L. J. Van Vliet, and K. Schutte, "Robust fusion of irregularly sampled data using adaptive normalized convolution," *EURASIP J. Appl. Signal Process.*, vol. 2006, p. 236, Feb. 2006.
[25] J. Mairal, F. Bach, J. Ponce, G. Sapiro, and A. Zisserman, "Non-local sparse models for image restoration," in *Proc. ICCV*, 2009, pp. 2272–2279.
[26] M. Protter and M. Elad, "Super resolution with probabilistic motion estimation," *IEEE Trans. Image Process.*, vol. 18, no. 8, pp. 1899–1904, Aug. 2009.
[27] P. Chatterjee and P. Milanfar, "A generalization of non-local means via kernel regression," in *Proc. SPIE Conf. Comput. Imaging*, 2008, pp. 1–9.
[28] H. Chang, D.-Y. Yeung, and Y. Xiong, "Super-resolution through neighbor embedding," in *Proc. IEEE CVPR*, 2004, pp. I-275–I-282.
[29] 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.
[30] A. Danielyan, A. Foi, V. Katkovnik, and K. Egiazarian, "Image and video super-resolution via spatially adaptive block-matching filtering," in *Proc. Int. Workshop Local Non-Local Approx. Image Process.*, 2008, pp. 1–8.
[31] R. Neelamani, H. Choi, and R. Baraniuk, "ForWaRD: Fourier-wavelet regularized deconvolution for ill-conditioned systems," *IEEE Trans. Signal Process.*, vol. 52, no. 2, pp. 418–433, Feb. 2004.
[32] H. Takeda, S. Farsiu, and P. Milanfar, "Deblurring using regularized locally adaptive kernel regression," *IEEE Trans. Image Process.*, vol. 17, no. 4, pp. 550–563, Apr. 2008.
[33] K. V. Suresh, G. Mahesh Kumar, and A. N. Rajagopalan, "Superresolution of license plates in real traffic videos," *IEEE Trans. Intell. Transp. Syst.*, vol. 8, no. 2, pp. 321–331, Jun. 2007.
[34] R. Tsai and T. S. Huang, "Multiframe image restoration and registration," in *Proc. Adv. Comput. Vis. Image Process.*, 1984, vol. 1, pp. 317–339.
[35] S. Farsiu, D. Robinson, M. Elad, and P. Milanfar, "Fast and robust multiframe super-resolution," *IEEE Trans. Image Process.*, vol. 13, no. 10, pp. 1327–1344, Oct. 2003.
[36] P. Getreuer, 2012. [Online]. Available: http://www.getreuer.info/home/tvreg
[37] C. Barnes, E. Shechtman, and A. Finkelstein, "PatchMatch: A randomized correspondence algorithm for structural image editing," *ACM Trans. Graph.– Proc. SIGGRAPH*, vol. 28, no. 3, p. 24, Aug. 2009.
[38] A. Adams, N. Gelfand, J. Dolson, and M. Levoy, "Gaussian KD-trees for fast high-dimensional filtering," *ACM Trans. Graph.–SIGGRAPH*, vol. 28, no. 3, p. 21, Aug. 2009.
[39] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: From error visibility to structural similarity," *IEEE Trans. Image Process.*, vol. 13, no. 4, pp. 600–612, Apr. 2004.
[40] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, "Image denoising by sparse 3D transform-domain collaborative filtering," *IEEE Trans. Image Process.*, vol. 16, no. 8, pp. 2080–2095, Aug. 2007.
[41] K. Dabov, A. Foi, V. Katkovnik, and K. O. Egiazarian, "Image restoration by sparse 3D transform-domain collaborative filtering," in *Proc. Image Process.: Algorithms Syst.*, 2008, p. 681 207.
[42] K. I. Kim and Y. Kwon, "Example-Based Learning for Single-Image Super-Resolution and jpeg Artifact Removal," Max Planck Inst. Biol. Cybern., Tübingen, Germany, Tech. Rep., 2008.
[43] R. Fattal, "Image upsampling via imposed edge statistics," *ACM Trans. Graph.–SIGGRAPH*, vol. 26, no. 3, p. 95, Jul. 2007.
[44] H. Zhang, J. Yang, Y. Zhang, and T. S. Huang, "Multi-scale non-local kernel regression for super resolution," in *Proc. IEEE ICIP*, 2011, pp. 1353–1356.
[45] C. Barnes, E. Shechtman, and A. Finkelstein, "The generalized PatchMatch correspondence algorithm," in *Proc. ECCV*, 2010, pp. 29–43.

Haichao Zhang (S'10) received the B.Eng. degree in computer science from Northwestern Polytechnical University, Xi'an, China, in 2007, where he is currently working toward the Ph.D. degree in the School of Computer Science.

His research interests include image processing, computer vision, machine learning, sparse representation, and its applications in pattern recognition and signal processing.

Mr. Zhang was a recipient of the Best Student Paper Award at the International Conference on Computer Vision in 2011, with Jianchao Yang and others, and the Honorable Mention Student Paper Award at the 2011 International Conference on Information Fusion.

Jianchao Yang (M'12) received the B.E. degree from the Department of Electronics Engineering and Information Science, University of Science and Technology of China, Beijing, China, in 2006 and the M.S. and Ph.D. degrees in electrical and computer engineering from the University of Illinois, Urbana, in 2011.

He is currently a Research Scientist with the Advanced Technology Laboratory, Adobe Systems Inc., San Jose, CA. His research interests include object recognition, face recognition, sparse coding, compressive sensing, image and video enhancement, denoising, and deblurring.

Yanning Zhang (SM'12) received the B.Eng. degree from Dalian University of Technology, Dalian, China, and the Ph.D. degree from the School of Marine Engineering, Northwestern Polytechnical University, Xi'an, China, in 1996.

She is currently a Professor and the Executive Dean of the School of Computer Science, Northwestern Polytechnical University. Her research interests include computer vision and pattern recognition, image and video processing, and intelligent information processing.

Dr. Zhang was the Organization Chair of the Asian Conference on Computer Vision 2009 and served as the Program Committee Chair of several international conferences.

Thomas S. Huang (LF'01) received the B.S. degree in electrical engineering from National Taiwan University, Taipei, Taiwan, in 1956 and the M.S. (in 1960) and Sc.D. degrees in electrical engineering from the Massachusetts Institute of Technology (MIT), Cambridge.

He was on the Faculty of the Department of Electrical Engineering, MIT, from 1963 to 1973 and on the Faculty of the School of Electrical Engineering and the Director of its Laboratory for Information and Signal Processing at Purdue University, West Lafayette, IN, from 1973 to 1980. In 1980, he joined the University of Illinois, Urbana, where he is currently a William L. Everitt Distinguished Professor of electrical and computer engineering, a Research Professor at the Coordinated Science Laboratory, and a Technology and Cochair of the Institute's major research theme Human Computer Intelligent Interaction at the Beckman Institute for Advanced Science. He has published 21 books and over 600 papers in network theory, digital filtering, image processing, and computer vision. His professional interests lie in the broad area of information technology, particularly the transmission and processing of multidimensional signals.

Dr. Huang is a member of the National Academy of Engineering, a member of the Academia Sinica, Republic of China, a foreign member of the Chinese Academies of Engineering and Sciences, and a fellow of the International Association of Pattern Recognition of the IEEE and the Optical Society of America. Among his many honors and awards are the Honda Lifetime Achievement Award, IEEE Jack Kilby Signal Processing Medal, and the KS Fu Prize of the International Association for Pattern Recognition.