Non-Local Kernel Regression for Image and Video Restoration

Conference Paper · September 2010

DOI: 10.1007/978-3-642-15558-1_41 · Source: DBLP

4 authors, including:

Haichao Zhang
Baidu Research
46 PUBLICATIONS **1,114** CITATIONS
SEE PROFILE

Jianchao Yang
Adobe Systems Inc
98 PUBLICATIONS **12,990** CITATIONS
SEE PROFILE

Non-Local Kernel Regression for Image and Video Restoration

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

Abstract

This paper presents a non-local kernel regression (NL-KR) model for various image and video restoration tasks, which exploits both the non-local self-similarity and local structural regularity properties in natural images. The non-local 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. Explicitly unifying both properties, our proposed non-local kernel regression framework is more robust in image estimation and the algorithm is applicable to various image and video restoration tasks. In this work, we apply the proposed model to image and video denoising, deblurring and super-resolution reconstruction. Extensive experimental results on both single images and realistic video sequences demonstrate the superiority of the proposed framework for desnoising, deblurring and super-resolution tasks over previous works both qualitatively and quantitatively.

Index Terms

local structural regression, non-local self-similarity, denoising, deblurring, super-resolution, restoration.

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],

Haichao Zhang and Yanning Zhang are with School of Computer Science, Northwestern Polytechnical University, Xi'an, 710129 China e-mail: hczhang@mail.nwpu.edu.cn, ynzhang@nwpu.edu.cn

Jianchao Yang and Thomas S. Huang are with Beckman Institute, Department of Electrical and Computer Engineering, University of Illinois at Urbana-Champaign, IL, 61801 USA e-mail: {jyang29, huang}@ifp.uiuc.edu.

Manuscript received January **, 2011; revised **** **, 2011.

sparse representation methods [3], [4], manifold methods [5], [6], etc. The success of such 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 self-similarity of local patterns.

Many conventional image processing algorithms are based on the assumption of local structural regularity, which says that there are meaningful structures in the spatial space of natural images. Examples are bilateral filtering [7] and structure tensor based methods [1], [8], [2], [9], [10]. 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 [7], 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, Biliteral 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 *et al.* [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 2D image plane, Li [8] and Takeda *et al.* [2] proposed respectively to improve the regression performance via regression kernels adapted to the local structures in the image. Li [9] further developed an implicit mixture motion model for video processing, which exploits the local spatial-temporal structures existing in videos. The generalization of 2-dimensional kernel regression to 3-dimensions has also been studied in [10] for video super-resolution. 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 methods exploiting the image self-similarity in natural images and videos are 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 non-local self-similarity has been widely used in texture synthesis literatures [11], 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 Non-Local Means (NL-Means) method [12]. Different from the local kernel regression method, NL-Means method breaks the locality constraint in the conventional restoration methods, and estimate the pixel value from all the similar patches collect from a large region. Its 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 multi-frame super-resolution tasks in [13]. Also, this self-similarity property

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

is thoroughly explored by Glasner *et al.* in [14] for addressing single image super-resolution problems and general inverse problems in [15] by Peyré *et al.*.

Image and video restoration aims to estimate the high-quality version of the low-quality observations, which are typically noisy and of low resolution. Figure 1 depicts the typical image formation process assumed in 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:

$$Y_k = DH_kX_k + \epsilon_k, \quad k = 1, 2, \cdots \tag{1}$$

where X_k represents the latent clean and sharp image while Y_k denotes the corresponding noisy, blurry and low-resolution observation. 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, or the motion blur caused by the relative motion between the imaging system and the target object. k is the frame/time-instance index. When $X_k = X, \forall k = 1, 2, \cdots$, Eq.(1) models the formation process of multiple observations of the same static scene. D denotes the subsampling operator modeling the finite sampling property of practical imaging systems. ϵ_k is usually assumed as an additive Gaussian noise term. In our preliminary work [16], we proposed a Non-Local Kernel Regression (NL-KR) method for image and video super-resolution (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 super-resolution. In our NL-KR model, we take advantage of both local structural regularity and non-local similarity in a unified framework for a more reliable and robust estimation. The non-local similarity and local structural regularity are intimately related, and are also complimentary in the sense that non-local similar pattern fusion can be regularized by the structural regularity while the redundancy from similar patterns enables a more accurate estimation for structural regression. Figure 2 gives a simplified schematic illustration of the proposed model.

The rest of the paper is organized as follows. We first review and summarize some related previous works for image and video restoration 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 as well as *state-of-the-art* methods. Section VI provides some discussions and concludes our paper.

II. Image Restoration: Prior Art

In this work, we are interested in image and video restorations where we desire to estimate the underlying latent clean and sharp image given the low quality observation(s). Examples are image denoising, deblurring and super-resolution. This section presents a brief technical review of local structural regression or filtering method as well as the non-local similarity-based approach, which are most closely related to our work.

The task of image restoration is to estimate the latent images $\{X_k\}$ given the low quality observations $\{Y_k\}$. 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 2D image plane and use y_i as a shorthand for $Y(\mathbf{x}_i)$, i.e., the pixel value of 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:

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

where $\mathbf{x}_i$ denotes the i-th 2D location on the image plane, $\mathcal{N}(\mathbf{x}_i)$ 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 non-similar pixels. Regarding $z(\mathbf{x}_i)$ as some regression function on the coordinate vector $\mathbf{x}_i$, Eq.(2) is essentially a zero-order estimation, where we directly minimize the weighted distance between $z(\mathbf{x}_i)$ 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. Specifically,

assuming the image is locally smooth to some order, we can rely on the local expansion of the function using Taylor series:

$$\begin{aligned} z(\mathbf{x}_i) =& z(\mathbf{x}) + \{\nabla z(\mathbf{x})\}^\top (\mathbf{x}_i - \mathbf{x}) \\ &+ \frac{1}{2}(\mathbf{x}_i - \mathbf{x})^\top \{\mathbf{H} z(\mathbf{x})\}(\mathbf{x}_i - \mathbf{x}) + \cdots \\ =& z(\mathbf{x}) + \{\nabla z(\mathbf{x})\}^\top (\mathbf{x}_i - \mathbf{x}) \\ &+ \frac{1}{2}\mathrm{vec}^\top \{\mathbf{H} z(\mathbf{x})\}\ \mathrm{vec}\{(\mathbf{x}_i - \mathbf{x})(\mathbf{x}_i - \mathbf{x})^\top\} + \cdots \end{aligned} \tag{3}$$

where ∇ and $\mathbf{H}$ denote the gradient and Hessian operators. $\mathrm{vec}(\cdot)$ is the operator which vectorizes a lexicographically a matrix into a vector. Eq.(3) can be further written as:

$$z(\mathbf{x}_i) = a_0 + \mathbf{a}_1^\top (\mathbf{x}_i - \mathbf{x}) + \mathbf{a}_2^\top \mathbf{tril}\{(\mathbf{x}_i - \mathbf{x})(\mathbf{x}_i - \mathbf{x})^\top\} + \cdots \tag{4}$$

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

$$\begin{aligned} \mathbf{a}_1 &= \nabla z(\mathbf{x}) = \left[\frac{\partial z(\mathbf{x})}{\partial x_1}, \frac{\partial z(\mathbf{x})}{\partial x_2}\right]^\top, \\ \mathbf{a}_2 &= \frac{1}{2}\left[\frac{\partial^2 z(\mathbf{x})}{\partial x_1^2}, 2\frac{\partial^2 z(\mathbf{x})}{\partial x_1 \partial x_2}, \frac{\partial^2 z(\mathbf{x})}{\partial x_2^2}\right]^\top. \end{aligned} \tag{5}$$

Therefore, the optimal regression coefficients $\{\mathbf{a}_k\}_{k=0}^K$ can be estimated via:

$$\begin{aligned} \{\mathbf{a}_k\}_{k=0}^K = \arg \min_{\{\mathbf{a}_k\}_{k=0}^K} \sum_{i=1}^N [& y_i - a_0 - \mathbf{a}_1^\top (\mathbf{x}_i - \mathbf{x}) \\ & - \mathbf{a}_2^\top \mathbf{tril}\{(\mathbf{x}_i - \mathbf{x})(\mathbf{x}_i - \mathbf{x})^\top\} - \cdots]. \end{aligned} \tag{6}$$

To reflect the differences of the contributions of all the pixels associated with the regression, a kernel can be added to Eq.(6)

$$\begin{aligned} \{\mathbf{a}_k\}_{k=0}^K = \arg \min_{\{\mathbf{a}_k\}_{k=0}^K} \sum_{i=1}^N [& y_i - a_0 - \mathbf{a}_1^\top (\mathbf{x}_i - \mathbf{x}) \\ & - \mathbf{a}_2^\top \mathbf{tril}\{(\mathbf{x}_i - \mathbf{x})(\mathbf{x}_i - \mathbf{x})^\top\} - \cdots] K_{\mathbf{x}_i}(\mathbf{x}_i - \mathbf{x}). \end{aligned} \tag{7}$$

It is easy to show that Eq.(7) can be reformulated into a matrix form as a weighted least-squares optimization problem,

$$\begin{aligned} \hat{\mathbf{a}} &= \arg\min_{\mathbf{a}} E(\mathbf{a}) \\ &= \arg\min_{\mathbf{a}} \|R_{\mathbf{x}_i} Y - \Phi \mathbf{a}\|^2_{W_{K_{\mathbf{x}_i}}} \\ &= \arg\min_{\mathbf{a}} (R_{\mathbf{x}_i} Y - \Phi \mathbf{a})^\top W_{K_{\mathbf{x}_i}} (R_{\mathbf{x}_i} Y - \Phi \mathbf{a}), \end{aligned} \tag{8}$$

where $R_{\mathbf{x}_i}$ is an operator which extracts the neighboring pixels centered at $\mathbf{x}_i$ (typically a patch) from Y and represents it as a vector. $W_{K_{\mathbf{x}_i}}$ is the weight matrix induced from the kernel

$$W_{K_{\mathbf{x}_i}} = \begin{bmatrix} K_{\mathbf{x}_i}(\mathbf{x}_1 - \mathbf{x}_i) & 0 & \cdots & 0 \\ 0 & K_{\mathbf{x}_i}(\mathbf{x}_2 - \mathbf{x}_i) & \cdots & \cdots \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & K_{\mathbf{x}_i}(\mathbf{x}_m - \mathbf{x}_i) \end{bmatrix}$$

with $m = |\mathcal{N}(\mathbf{x}_i)|$, and Φ is the polynomial basis from Taylor expansion,

$$\Phi = \begin{bmatrix} 1 & (\mathbf{x}_1 - \mathbf{x}_i)^\top & \textbf{tril}\{(\mathbf{x}_1 - \mathbf{x}_i)(\mathbf{x}_1 - \mathbf{x}_i)^\top\}^\top & \cdots \\ 1 & (\mathbf{x}_2 - \mathbf{x}_i)^\top & \textbf{tril}\{(\mathbf{x}_2 - \mathbf{x}_i)(\mathbf{x}_2 - \mathbf{x}_i)^\top\}^\top & \cdots \\ \vdots & \vdots & \vdots & \vdots \\ 1 & (\mathbf{x}_m - \mathbf{x}_i)^\top & \textbf{tril}\{(\mathbf{x}_m - \mathbf{x}_i)(\mathbf{x}_m - \mathbf{x}_i)^\top\}^\top & \cdots \end{bmatrix} \tag{9}$$

where **tril**$(\cdot)$ extracts the lower triangular part of a matrix and stack it to a column vector. It is also easy to see that the regression coefficient vector with respect to Φ is:

$$\mathbf{a} = [a_0, \mathbf{a}_1^\top, \mathbf{a}_2^\top, \cdots]^\top. \tag{10}$$

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

$$\frac{\partial E(\mathbf{a})}{\partial \mathbf{a}} = 2\Phi^\top W_{K_{\mathbf{x}_i}}(\Phi\mathbf{a} - R_{\mathbf{x}_i}Y), \tag{11}$$

and set it to be zero, we have

$$\hat{\mathbf{a}} = \left[\Phi^\top W_{K_{\mathbf{x}_i}}\Phi\right]^{-1}\Phi^\top W_{K_{\mathbf{x}_i}}R_{\mathbf{x}_i}Y. \tag{12}$$

According to the regression basis defined in Eq.(9) and also Eq.(10), the first element of the regression coefficient vector $\hat{\mathbf{a}}$ is the desired pixel value estimation at $\mathbf{x}_i$, therefore,

$$\begin{aligned} \hat{z}(\mathbf{x}_i) = \hat{a}_0 &= \mathbf{e}_1^\top\hat{\mathbf{a}} \\ &= \mathbf{e}_1^\top\left[\Phi^\top W_{K_{\mathbf{x}_i}}\Phi\right]^{-1}\Phi^\top W_{K_{\mathbf{x}_i}}R_{\mathbf{x}_i}Y, \end{aligned} \tag{13}$$

where $\mathbf{e}_1$ is a vector with the first element equal to one, and the rest zero. This formulation for local structural regression is a general framework, including example works of Gaussian filtering [17], Bilateral filtering [7], and structure kernel guided filtering [1] [2] as special cases.

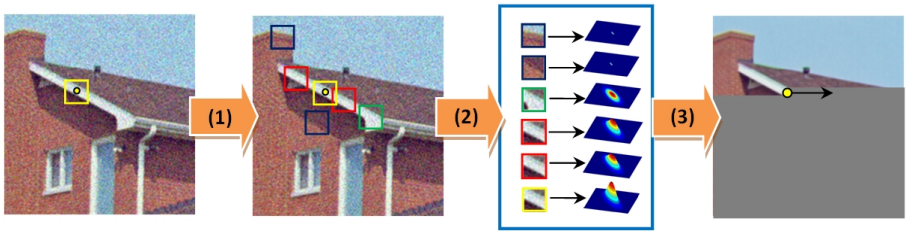

Fig. 2. **Schematic illustration of Non-Local Kernel Regression framework.** (1) Similar patch searching: different colors indicate the similarity (red the highest, green the medium and blue the least); (2) Structural kernel estimation and reweighting: estimate a regression kernel adapted to the structure at each position where the similar patches reside and re-weight them according to similarity; (3) Non-local kernel regression: estimate the value for the query point with both local structural and non-local similar information in raster-scan order.

B. Non-Local 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 [11], image inpainting [18], denoising [12] [19] and super-resolution [13] [20]. 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 *et al.* in [12] for image denoising has become very popular, due to its effectiveness despite of 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 not only their own pixel values but also their local neighboring pixels,

$$z(\mathbf{x}_i) = \frac{\sum_{j\in\mathcal{P}(\mathbf{x}_i)} w_{ij} y_j}{\sum_{j\in\mathcal{P}(\mathbf{x}_i)} w_{ij}} \tag{14}$$

where $\mathcal{P}(\mathbf{x}_i)$ denotes the index set for similar pixel observations for $\mathbf{x}_i$ (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 $\mathbf{x}_j$ [12]. Eq.(14) can again be reformulated into a least-squares optimization

problem as follows:

$$\begin{aligned}\hat{z}(\mathbf{x}_i) &= \arg\min_{z_i} E(z_i) \\ &= \arg\min_{z_i} \sum_{j\in\mathcal{P}(\mathbf{x}_i)} \left[y_j - z(\mathbf{x}_i)\right]^2 w_{ij} \\ &= \arg\min_{z_i} \|\mathbf{y} - \mathbf{1}z(\mathbf{x}_i)\|^2_{W_{\mathbf{x}_i}} \\ &= \arg\min_{z_i} \left(\mathbf{y} - \mathbf{1}z(\mathbf{x}_i)\right) W_{\mathbf{x}_i} \left(\mathbf{y} - \mathbf{1}z(\mathbf{x}_i)\right)\end{aligned} \tag{15}$$

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

$$W_{\mathbf{x}_i} = \begin{bmatrix} w_{i1} & 0 & \cdots & 0 \\ 0 & w_{i2} & \cdots & \cdots \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & w_{im} \end{bmatrix}, \tag{16}$$

where $m = |\mathcal{P}(\mathbf{x}_i)|$. To show that Eq.(14) is the solution to Eq.(15), we differentiate $E(z_i)$ with respect to z_i,

$$\frac{\partial E(z_i)}{\partial z_i} = 2 \times \mathbf{1}^\top W_{\mathbf{x}_i}(\mathbf{1}z(\mathbf{x}_i) - \mathbf{y}). \tag{17}$$

Set it to be zero and solve for z_i, we get:

$$z(\mathbf{x}_i) = \left[\mathbf{1}^\top W_{\mathbf{x}_i}\mathbf{1}\right]^{-1} \mathbf{1}^\top W_{\mathbf{x}_i}\mathbf{y}. \tag{18}$$

Since $W_{\mathbf{x}_i}$ is a diagonal matrix, thus Eq.(14) and Eq.(18) are equal.

Compared with Eq.(8), the NL-Means estimation Eq.(15) 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 non-local fashion, instead of basing on the spatial kernel as before. As the weight matrix is diagonal, the NL-Means algorithm means that the pixels within a local neighborhood do not contribute directly to the estimation of the current pixel value.

III. Non-Local Kernel Regression Model

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

A. Mathematical Formulation

In this subsection, we derive our Non-Local Kernel Regression (NL-KR) algorithm, which makes full use of both cues from *local* structural regularity and *non-local* 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 non-local similarity search; and non-local similarity provides the redundancy preventing possible overfitting of the local structural regression. Instead of using a *point prediction* model in non-local 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 can try to make use of all the *non-local similar* patches in natural images. Mathematically, the proposed high-order Non-Local Kernel Regression model is formulated as:

$$\begin{aligned}\hat{\mathbf{a}} &= \arg\min_{\mathbf{a}} E(\mathbf{a}) \\ &= \arg\min_{\mathbf{a}} \frac{1}{2}\overbrace{w_{ii}\|R_{\mathbf{x}_i}Y - \Phi\mathbf{a}\|^2_{W_{K_{\mathbf{x}_i}}}}^{local} + \frac{1}{2}\overbrace{\sum_{j\in\mathcal{P}(\mathbf{x}_i)\setminus\{i\}} w_{ij}\|R_{\mathbf{x}_j}Y - \Phi\mathbf{a}\|^2_{W_{K_{\mathbf{x}_j}}}}^{non-local} \\ &= \arg\min_{\mathbf{a}} \frac{1}{2}\sum_{j\in\mathcal{P}(\mathbf{x}_i)} w_{ij}\|R_{\mathbf{x}_j}Y - \Phi\mathbf{a}\|^2_{W_{K_{\mathbf{x}_j}}} \\ &= \arg\min_{\mathbf{a}} \frac{1}{2}\sum_{j\in\mathcal{P}(\mathbf{x}_i)} \|R_{\mathbf{x}_j}Y - \Phi\mathbf{a}\|^2_{\tilde{W}_{\mathbf{x}_j}}\end{aligned} \tag{19}$$

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 similar patch index set for $\mathbf{x}_i$. w_{ij} is calculated between the location $\mathbf{x}_i$ of interest and position $\mathbf{x}_j$ ($j \in \mathcal{P}(\mathbf{x}_i)$) by measuring the similarity of their neighborhoods weighted by a Gaussian kernel:

$$w_{ij} = \exp\left(-\frac{\|R_{\mathbf{x}_i}Y - R_{\mathbf{x}_j}Y\|^2_{W_G}}{2\sigma^2}\right) \tag{20}$$

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 1. This term also contributes as a fidelity loss, as the estimation should be close to the observation.
2) **Non-local regression term**: instead of zero-order point estimation as in Non-Local means, higher-order kernel regression is also used to make full use the structural redundancy in the similar patches.

To get the regression coefficients, we differentiate the right hand side of Eq.(19) with respect to $\mathbf{a}$

$$\frac{\partial E(\mathbf{a})}{\partial \mathbf{a}} = \Phi^\top \sum_{j\in\mathcal{P}(\mathbf{x}_i)} w_{ij} W_{K_{\mathbf{x}_j}}(\Phi\mathbf{a} - R_{\mathbf{x}_j}Y). \tag{21}$$

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

$$\hat{\mathbf{a}} = \left[\Phi^\top \left(\sum_{j \in \mathcal{P}(\mathbf{x}_i)} w_{ij} W_{K_{\mathbf{x}_j}} \right) \Phi \right]^{-1} \Phi^\top \sum_{j \in \mathcal{P}(\mathbf{x}_i)} w_{ij} W_{K_{\mathbf{x}_j}} R_{\mathbf{x}_j} Y. \tag{22}$$

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

Examination on Eq.(22), we have the following two comments:

- The structural kernel is estimated from the contaminated observations, and thus is not robust. Compared with Eq.(13), with non-local 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.
- Compared with Eq.(18), our model can regularize the estimation from the non-local patches by structural higher-order 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 structure and non-local 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 gives smaller weights to the pixels far away. While achieving reasonable results on flat areas, such a kernel suffers from the severe limitation that it usually blurs the edge structures in the image, as their profile supports are usually smaller than that of the isotropic Gaussian kernel. Bilateral filtering improves over isotropic Gaussian kernel by introducing the pixel value into the weight calculation other than the spatial coordinates, which shows to preserve the edges structures better. Moreover, it is desirable to make the kernel adaptive to the image structures, which can be achieved via the local gradient analysis [2], [21]. Given the observation Y and a query location $\mathbf{x}_i$, we can construct a structure adaptive kernel as:

$$K_{\mathbf{x}_i}(\mathbf{x} - \mathbf{x}_i) = \frac{1}{\sqrt{\det(T)}} \exp\left\{ -\frac{1}{2} (\mathbf{x} - \mathbf{x}_i)^\top T^{-1} (\mathbf{x} - \mathbf{x}_i) \right\}, \tag{23}$$

where T is the diffusion tensor at $\mathbf{x}_i$ controlling the spatial structure of the kernel [21]. Given two unit vectors $\mathbf{u}$ and $\mathbf{v}$ defined by the gradient and tangent direction respectively, we can construct the diffusion tensor as:

$$T = f \mathbf{u}\mathbf{u}^\top + g \mathbf{v}\mathbf{v}^\top. \tag{24}$$

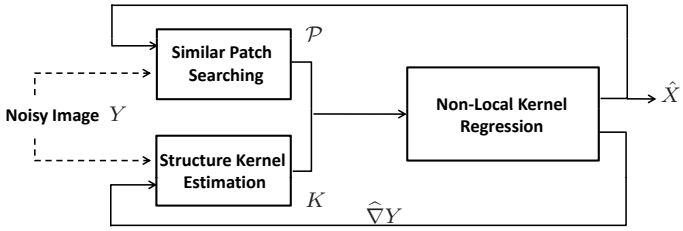

Fig. 3. **A schematic illustration of the iterative NL-KR framework.** Given a noisy input Y, the proposed method estimates the underlying noise-free data iteratively.

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:

$$f(\alpha, \beta) = \frac{\beta + \gamma}{\alpha + \gamma}, \quad g(\alpha, \beta) = \frac{\alpha + \gamma}{\beta + \gamma} \quad (\gamma \geq 0), \tag{25}$$

where α and β are the eigen values of the structure tensor [21], reflecting the strength of the gradient along each eigenvector directions. α, β, $\mathbf{u}$ and $\mathbf{v}$ can be calculated from the following relation using singular value decomposition (SVD):

$$\nabla Y_{\mathbf{x}_i} \nabla Y_{\mathbf{x}_i}^{\top} = \alpha \mathbf{u}\mathbf{u}^{\top} + \beta \mathbf{v}\mathbf{v}^{\top} \tag{26}$$

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

C. Iterative Non-Local Kernel Regression

The structure kernel is data dependent, which manifests the non-liner 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 Figure 3. Although without an analysis of optimality and convergence, we experimentally validate such an iterative procedure on image denoising in subsection V-A, where more iterations which apply the iterative refined structural kernels indeed bring more aggressive noise removal.

D. Relations to Other Works

Tons of works have emerged recently based on non-local redundancy and local regressions for image and video processing. It is worthwhile to talk about the relations of the proposed model to those previously

[1] One can refer to [21] for other choices of diffusion tensor.

proposed algorithms. The non-local models in [22], [12], and [13] use the redundancy from non-local self-similarity, but do not include the spatial structure explicitly as a regularization. The high order generalization of non-local means in [23] uses the computation of non-local similarity to find the local kernel for regression, which actually violates the philosophy of the non-local model. Local structural regression models [1] [2] [7] explicitly employ the spatial kernel for regularization, but neglect the redundancy of similar local patterns useful for robust estimation. The 3D kernel regression method [10] exploits the local spatial-temporal structure by extending their 2D spatial kernel regression, also discards the non-local self-similarity. Sparse representation for denoising [3] and super-resolution [4] do local regression using bases learned from a training database, which performs estimation on each individual local patch and discard the patch redundancy. The sparse representation model is later generalized in [19] for image denoising by performing simultaneous sparse coding over similar patches found in different locations of the image. However, the non-local redundancy is used in a hard assignment clustering way instead of a soft way. [14] fully explores the self-similarity property for single image super-resolution, 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 above is a general model that can be applied to many image and video restoration tasks. In this work, we apply our model to denoising, deblurring and super-resolution 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 single observation:

$$Y = X + \epsilon. \tag{27}$$

To denoise the noisy observation Y, lots of different approaches have been developed. They mainly fall into one of the two categories as we 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], [7], [8]. The denoising methods by learning an over-complete dictionary using image patches have shown to achieve promising

Algorithm 1 (Non-local kernel regression for image denoising).

1: **Input:** a noisy image Y, number of iterations T.

2: **Initialize** the current denoising estimation $X^0 = Y$, estimate the image gradient ∇Y using X^0;

3: **For** $t = 1, 2, \cdots, T$, 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 X^{t-1};
- Calculate the diffusion tensor $K_{\mathbf{x}_i}$ use Eq.(26) and Eq.(23);
- 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 Eq.(22) and update the current estimation of X^t at $\mathbf{x}_i$ with $X^t(\mathbf{x}_i) = \mathbf{e}_1^\top \hat{\mathbf{a}} = \hat{\mathbf{a}}(1)$;
- Update the image gradient ∇Y at $\mathbf{x}_i$ as $\nabla Y_{\mathbf{x}_i} = [\hat{\mathbf{a}}(2), \hat{\mathbf{a}}(3)]^\top$.

5: **End**

6: **End**

7: **Output:** a denoised image $\hat{X} = X^T$.

results for denoising [24]. The second category includes the recent non-local denoising methods [12], [25]. These methods break the conventional locality constraint for estimation, which perform estimation based on similar patches collected in a non-local fashion.

For image denoising, our NL-KR model (19) developed above can be applied directly, by constructing the similar patch set $\mathcal{P}$ using the noisy observation Y, thus combining the local structural and non-local 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 depicted in Figure 3.

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

$$Y = HX + \epsilon. \tag{28}$$

Eq.(28) can be rewritten equivalently in the frequency domain as:

$$\mathcal{Y}(u,v) = \mathcal{H}(u,v)\mathcal{X}(u,v) + \mathcal{E}(u,v). \tag{29}$$

The most direct way of estimating $\mathcal{X}$ is through inverse filtering:

$$\hat{\mathcal{X}}(u,v) = \frac{\mathcal{Y}(u,v)}{\mathcal{H}(u,v)} = \mathcal{X}(u,v) + \frac{\mathcal{E}(u,v)}{\mathcal{H}(u,v)}. \tag{30}$$

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 (30) 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:

$$\begin{aligned} \hat{\mathcal{X}}(u,v) &= \frac{\mathcal{H}^*(u,v)\mathcal{Y}(u,v)}{|\mathcal{H}(u,v)|^2 + \lambda} \\ &= \frac{|\mathcal{H}(u,v)|^2}{|\mathcal{H}(u,v)|^2 + \lambda}\mathcal{X}(u,v) + \frac{\mathcal{H}^*(u,v)}{|\mathcal{H}(u,v)|^2 + \lambda}\mathcal{E}(u,v), \end{aligned} \tag{31}$$

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 [26]. Following the combined Fourier-Wavelet Regularized Deconvolution (ForWaRD) method developed in [26], we apply our NL-KR framework to image deblurring task as follows. Similar to 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 final estimation, our proposed method takes advantage both the local and non-local 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 [26]. Furthermore, iterative back-projection schemes as used in [27], [28] 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.

B. Super-resolution

Image super-resolution (SR) aims to estimate a high-resolution image (HR) from a single or a set of low-resolution (LR) observations. Conventional multi-frame SR follows the steps of (1) global motion

Algorithm 2 (Non-local kernel regression for image deblurring).

1: **Input:** a blurry image Y, the blur kernel h, regularization parameter λ.
2: **Initialize** frequency representation for Y and h as $\mathcal{Y}$ and $\mathcal{H}$ using FFT.
3: **Perform** Wiener filtering using Eq.(31) to generate $\hat{\mathcal{X}}$;
4: **Generate** the spatial domain estimation $\tilde{X}$ by applying inverse FFT to $\hat{\mathcal{X}}$;
5: **Perform** NL-KR denoising to $\tilde{X}$ to generate the final deblurring result $\hat{X}$;
6: **Output:** a deblurred image $\hat{X}$.

estimation, (2) image wrapping and (3) data fusion [29] [30]. These methods are limited in the assumed global motion model, and can not be applied to realistic videos that almost certainly contain arbitrary motion patterns. Recently, several multi-frame SR algorithms based on fuzzy motion estimation of local image patches are proposed to process real videos [10] [13]. We will show that similarly our model can also be applied to realistic videos, while achieving better results both qualitatively and quantitatively. Besides, 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 Eq.(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 Z_k followed by deblurring. As now we have two different spatial scales, i.e., high- and low-resolution image grids, the following notations are introduced for ease of presentation. We let r denote the zoom factor, $\mathbf{x}$ 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. Please refer to Figure 4 for 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.

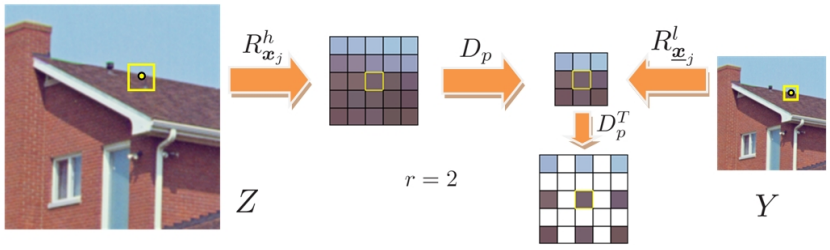

Fig. 4. **Operator Illustration.** $R^l_{\underline{\mathbf{x}}_j} Y$, the patch of the LR image Y at $\underline{\mathbf{x}}_j$, is formed by downsampling HR patch $R^h_{\mathbf{x}_j} Z$ by factor $r = 2$ as $D_p R^h_{\mathbf{x}_j} Z$, keeping the center pixel still in the center. Operator D_p^T up samples a patch with zero padding.

Using Eq.(1), (19), the NL-KR model adapted for SR tasks is formulated as:

$$\begin{aligned}\hat{\mathbf{a}} &= \arg\min_{\mathbf{a}} \frac{1}{2} \sum_{j \in \mathcal{P}(\mathbf{x}_i)} \|D_p^T (R^l_{\underline{\mathbf{x}}_j} Y - D_p \Phi \mathbf{a})\|^2_{\tilde{W}_{\mathbf{x}_j}} \\ &= \arg\min_{\mathbf{a}} \frac{1}{2} \sum_{j \in \mathcal{P}(\mathbf{x}_i)} (R^l_{\underline{\mathbf{x}}_j} Y - D_p \Phi \mathbf{a})^T D_p \tilde{W}_{\mathbf{x}_j} D_p^T (R^l_{\underline{\mathbf{x}}_j} Y - D_p \Phi \mathbf{a}) \\ &= \arg\min_{\mathbf{a}} \frac{1}{2} \sum_{j \in \mathcal{P}(\mathbf{x}_i)} \|R^l_{\underline{\mathbf{x}}_j} Y - D_p \Phi \mathbf{a}\|^2_{\tilde{W}^D_{\mathbf{x}_j}},\end{aligned} \tag{32}$$

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 $R^h_{\mathbf{x}_j} Z$ centered at query location $\mathbf{x}_j$ for the blurred HR image Z, and $\tilde{W}^D_{\mathbf{x}_j} = D_p \tilde{W}_{\mathbf{x}_j} D_p^T$. Solution to Eq.(32) is:

$$\hat{\mathbf{a}} = \left[\Phi^T \left(\sum_{j \in \mathcal{P}(\mathbf{x}_i)} D_p^T \tilde{W}^D_{\mathbf{x}_j} D_p \right) \Phi \right]^{-1} \Phi^T \sum_{j \in \mathcal{P}(\mathbf{x}_i)} D_p^T \tilde{W}^D_{\mathbf{x}_j} R^l_{\underline{\mathbf{x}}_j} Y. \tag{33}$$

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

As we can see, the missing pixels on the high resolution grid are filled up by multiple low resolution observations found in a non-local way on the current frame or current sequence. These low resolution 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 [31] in our experiment following [13], [20] for fair of comparison. Algorithm 3 describes the practical implementation for the proposed model.

V. Experimental Validation

Experiments on image and video denoising, deblurring and super-resolution tasks are carried out to verify the effectiveness of the proposed method in this section. In the experiments, the patch size is fixed as 5×5 and $\gamma = 1$ is used for diffusion tensor computation.

Algorithm 3 (Non-local kernel regression for image super-resolution).

1: **Input:** a low resolution video sequence $\underline{Y} = [Y_1, Y_2, ..., Y_M]$, zoom factor r.

2: **Initialize** an enlarged sequence $\underline{\tilde{Y}} = [\tilde{Y}_1, \tilde{Y}_2, ..., \tilde{Y}_M]$ with bicubic interpolation for all the low resolution frames.

3: **For** each pixel location $\mathbf{x}_i$ on the high resolution image grid for frame Y_m, do

- Construct the similar patch index set $\mathcal{P}(\mathbf{x}_i)$ with sequence $\underline{\tilde{Y}}$;
- Estimate the image gradient $\nabla Y_{\mathbf{x}_i}$;
- Calculate the diffusion tensor $K_{\mathbf{x}_i}$ use Eq.(26) and Eq.(23).

4: **End**

5: **For** each pixel location $\mathbf{x}_i$ on the high resolution image grid for frame Y_m, 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 Eq.(33) and update the current estimation of Z_m at $\mathbf{x}_i$ with $Z_m(\mathbf{x}_i) = \mathbf{e}_1^T \hat{\mathbf{a}}$.

6: **End**

7: **Perform** deblurring for Z_m: $X_m = \mathrm{TVdeblur}(Z_m)$.

8: **Output:** a high resolution video frame X_m.

A. Denoising Experiment

In this subsection, we apply the proposed NL-KR method to denoising applications. For denoising, we perform pixel-wise value estimation using NL-KR on the 2D 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 standard deviation of $\sigma = 25$ to the clean test image. Then we perform denoising using the different algorithms mentioned above. The results are summarized in Table I. Peak Signal to Noise Ratio (PSNR) is adopted as an evaluation metric for denoising results, which is defined as:

$$\mathrm{PSNR} = 10 \log_{10} \frac{255^2}{\sqrt{\frac{1}{N}\sum_i (\hat{X} - X)_i^2}},$$

where $\hat{X}$ denotes the denoised estimation and X the original clean image. N is the total number of pixels in X. The Structural similarity (SSIM) index [32] is also used for objective evaluation.

As can be seen from Table I, the proposed NL-KR method performs better than NL-Means [12] and Kernel Regression [2] on denoising tasks, which use only the non-local/local information for estimation

TABLE I

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

Image	NL-Means	KR	KSVD	BM3D	Proposed
Lena	30.3317	31.5595	31.3393	32.0387	31.8659
Barbara	28.4759	28.5306	29.5366	30.6430	29.6939
Boat	28.5003	29.1178	29.2142	29.8439	29.5152
Pepper	29.0695	29.2684	29.5394	30.2330	29.9370
House	30.6225	31.4873	32.1171	32.9189	32.4258
Cameraman	28.2604	27.8295	28.5236	29.1137	28.7876
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

respectively. This clearly verifies the effectiveness of exploiting both local structural regularity and the non-local self-similarity as in the proposed method. The performance of the proposed NL-KR method is also better than that of the KSVD method [24], which is a learning based denoising method and requires additional training phase before denoising. Moreover, the denoising result of our method is comparable with BM3D in terms of PSNR and SSIM (there is no visually significant differences between the results, see Figure 5), which is the current best denoising algorithm to the best of our knowledge. It is worth mentioning that BM3D has an additional step of Wiener filtering after its 3D collaborative filtering step to boost its denoising performance, while our proposed method without such an additional step can already generate comparable results to BM3D. The denoising results for 'House' image are shown in Figure 5. As can be seen from Figure 5, the denoising result from NL-Means method suffers from blocky artifacts, while the denoised image using KR method has flow-like effects. The denoising result using the proposed method is visually similar to the result of BM3D [33].

Another denoising experiment is carried out using a real noisy color image, as shown in Figure 6. As can be seen, the NL-KR method can generate comparable results with other *state-of-the-art* denoising methods. The absolute difference images between the denoised images and the noisy image are also shown in Figure 6. As we can see, the proposed method preserves the structures well while removing the noise effectively, which demonstrate the applicability of the proposed method for real image denoising.

Fig. 5. **Image denoising** (PSNR, SSIM in brackets). Left to right: Noisy image, NL-Means [12] (30.6225, 0.7678), KR [2] (31.4873, 0.8317), KSVD [24] (32.1171, 0.8445), BM3D [33] (32.9189, 0.8563) and NL-KR (32.4258, 0.8497).

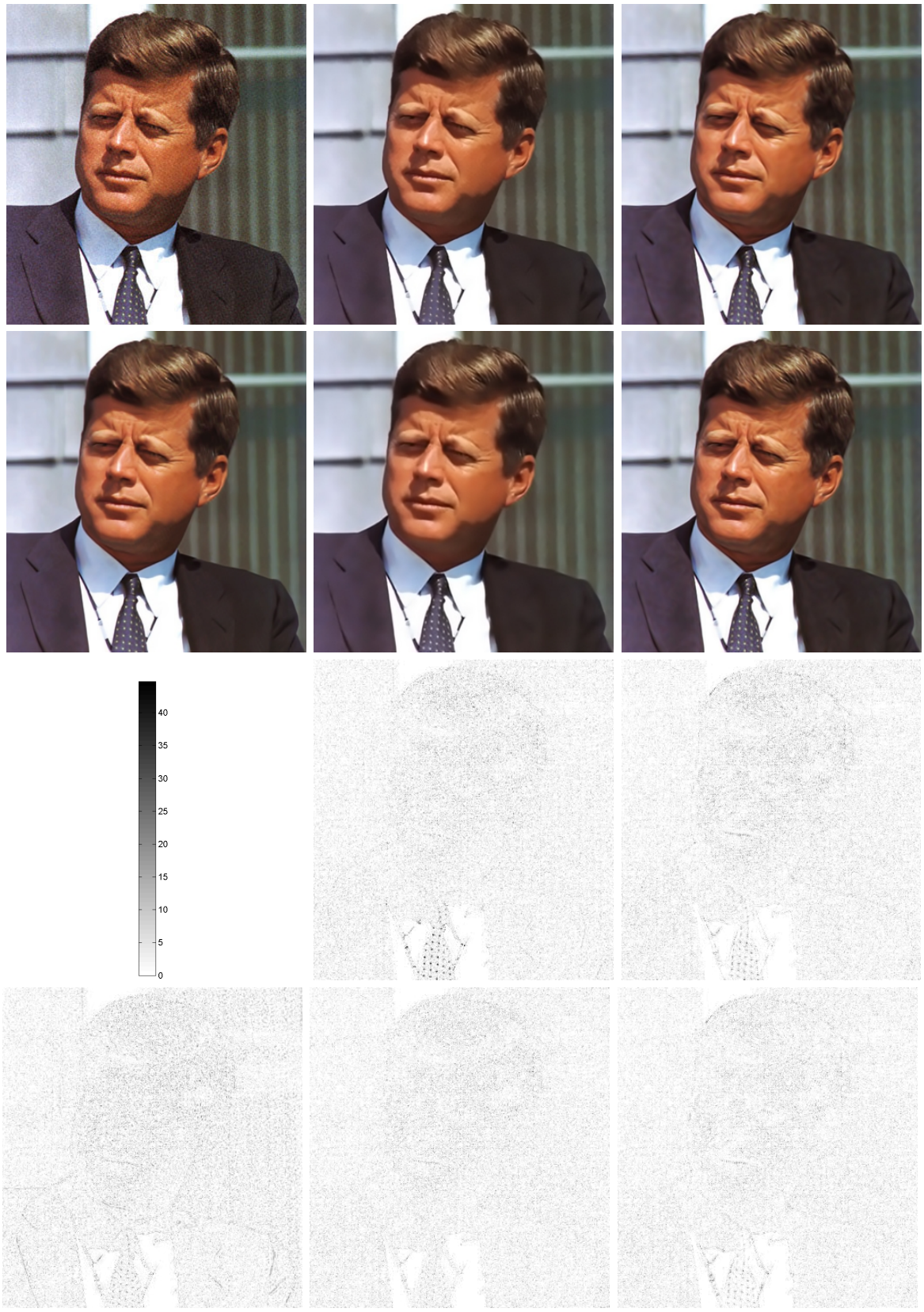

Fig. 6. **Image denoising for real noisy color images**. Left to right: Noisy image, NL-Means [12], KR [2], KSVD [24], BM3D [33] and NL-KR. The images at bottom are the corresponding absolute difference between each denoised image with respect to the noisy image.

B. Deblurring Experiment

In this subsection, 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. ISNR actually quantifies the difference of the SNR of the deblurred image $\hat{X}$ and the SNR of the blurry observation Y:

$$\mathrm{ISNR} = 10\log_{10}\frac{\sum_i (Y-X)_i^2}{\sum_i (\hat{X}-X)_i^2}. \tag{34}$$

Again, SSIM is also used as another metric for evaluation. The experimental setups are as follows. A 9×9 box blur is used for generating the blurry image. Then Gaussian noise is added to the blurry image with BSNR=40 dB,[2] following the settings in [26]. We compare the proposed method with two methods: classical Wiener filtering and ForWaRD [26], which is one of the best non-iterative deblurring methods.

The results are summarized in Table II. As can be seen from the results in Table II, the proposed method can generate better deblurring results than ForWaRD both in terms of ISNR as well as SSIM on all the test images. To further verify the effectiveness, we show some of the deblurred results in Figure 7. As can be seen from this figure, the Wiener filtering method suffers from noise amplification. 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 non-local similarity information, the proposed NL-KR method can avoid the noise amplification during deblurring while can retain the detail structures well.

C. Super-Resolution Experiment

The proposed NL-KR model can handle both single image and multiple frame SR naturally. In this subsection, we validate the performance of the proposed method with experiments on single images, synthetic and real video sequences. Performance comparisons are performed with related *state-of-the-art* algorithms. We use both PSNR and SSIM index [32] to evaluate different algorithms objectively.

In all the experiments, we focus on zooming the LR frame(s) by factor of 3. These LR frames are modeled by first blurring the HR frames with a 3×3 uniform PSF and downsampling with decimation factor of 3. Gaussian noise of standard deviation 2 is added to the LR frames to model the real imaging system. In our algorithm, the LR patch size is fixed as 5×5, and the corresponding HR patch size is

[2]Blurred signal-to-noise ratio, defined as $\mathrm{BSNR} = 10\log_{10}\frac{\sum_i (\epsilon)_i^2}{\sum_i (\hat{X}-X)_i^2}$.

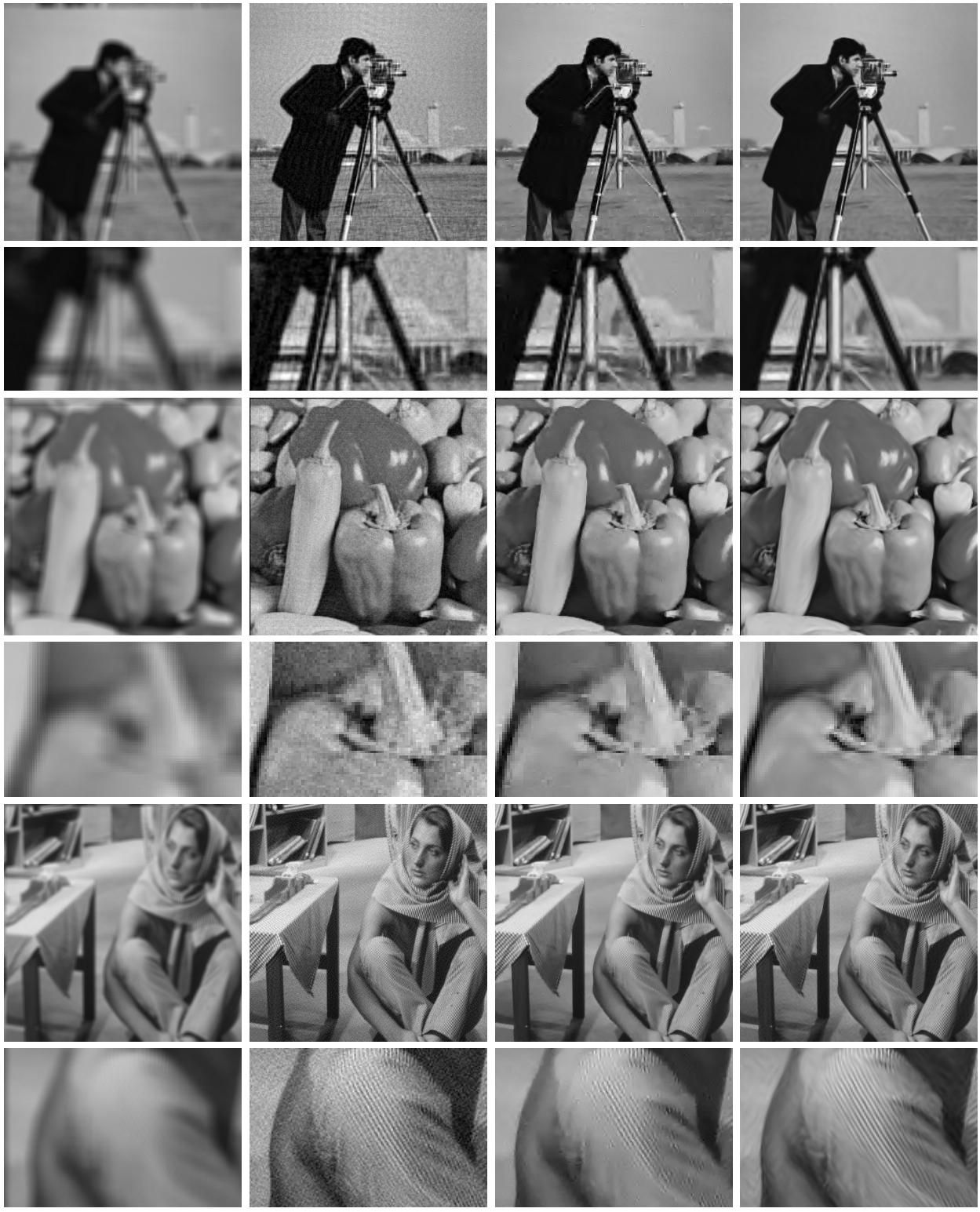

Fig. 7. **Image deblurring** (ISNR, SSIM for 'cameraman' image in brackets). Left to right: Blurry and noisy image, Wiener filtering (8.9954, 0.6364), ForWaRD [26] (10.7686, 0.8346) and NL-KR (10.9731, 0.8369).

TABLE II

DEBLURRING RESULTS IN TERMS OF ISNR (TOP) AND SSIM (BOTTOM)

Image	Wiener	ForWaRD	Proposed
Lena	4.0528	12.4329	13.1319
Barbara	4.8054	7.7710	8.6114
Boat	4.0141	11.1209	11.4537
Pepper	11.0192	13.9280	14.1066
House	9.0394	13.8176	14.3620
Cameraman	8.9954	10.7686	10.9731
Lena	0.4163	0.8694	0.8823
Barbara	0.5062	0.7807	0.8086
Boat	0.4773	0.8296	0.8379
Pepper	0.6922	0.8873	0.8945
House	0.6254	0.8680	0.8716
Cameraman	0.6364	0.8346	0.8369

thus 13×13. The support of the non-local similar patch searching is fixed to be the 10-nearest neighbors. We set $\sigma = 169c$ with $c = 0.06$ for similarity weight calculation. For image deblurring, we use a Total Variation based deblurring algorithm [31].

1) Single Frame based Super-Resolution: We will first evaluate the proposed method on single image SR task. In the first set of experiments, we specifically compare the proposed model with 2D case of Generalized NL-Means (GNL-Means) [13] and Kernel Regression (KR) [2] in order to show that our model is more reliable and robust for estimation. We take one frame from each of the popular test sequences: Foreman, Miss America and Suzie used in [13], 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 is constantly better than 2D GNL-Means and 2D Kernel Regression. The results of Nearest Neighbor interpolation (NN), Bicubic Interpolation (BI) and Sparse Coding (SC) method [4] are also provided as references. Figure 8 shows the visual quality comparisons on Foreman. As shown, the 2D GNL-Means method is prone to block artifacts due to poor patch matching within a single image and the 2D Kernel Regression method generates ghost effects due to insufficient observation for regularizing the local regression. Our result, however, is free of either of these artifacts.

We further make more comparisons with *state-of-the-art* methods on real images, where the input LR image is zoomed by a factor of 4, as shown in Figure 9 and Figure 10. Note that these methods are

Fig. 8. **Single-frame super resolution** ($\times 3$, PSNR, SSIM in brackets). Left to right: NN(28.6917, 0.8185), BI(30.9511, 0.8708), GNL-Means(31.9961, 0.8747)[13], KR(32.4479, 0.8862)[2], NL-KR(32.7558, 0.8918). GNL-Means generates block effect while KR generates ghost artifacts. Our method does not suffer from these problems.

designed specifically to work on single images. In Figure 9, it can be seen that the proposed method can preserve more details than Fattal's method [35] and is comparable with Kim's method [34] and the more recent work [14]. In Figure 10, however, our algorithm outperforms both [35] and [14], where our result is free of the *jaggy* artifacts on the edges and the characters generated by our method is more realistic. The improvement comparison could be more impressive if one notices that in [14], multiple scales are used for similar pattern matching while our method only uses one scale, although our results can be further improved by using multi-scale self-similarities.

2) Synthetic Experiment for Multi-Frame SR: Our second experiment is conducted on synthetic image frames. We generate 9 LR images from one HR image by blurring the HR image with a 3×3 uniform PSF and then decimate the blurred HR image every 3rd row or column with shifts of $\{0, 1, 2\}$ pixels. Gaussian noise with standard deviation of 2 is 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 is cited directly from [25], which are obtained from noise-free observations.

3) Evaluation on Real 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

Fig. 9. **Single-frame super resolution for real color images** ($\times 4$). From left to right: NN, BI, Kim's method [34], Fattal's method [35], Glasner's method [14], NL-KR. Note that our result preserves more details than Fattal's method and is comparable to results from Kim's learning based method and recently proposed method by Glasner.

TABLE III

PSNR (TOP) AND SSIM (BOTTOM) RESULTS FOR SINGLE IMAGE SUPER-RESOLUTION

Image	NN	BI	GNL-Means [13]	KR [2]	SC [4]	Proposed
Foreman	28.6917	30.9511	31.9961	32.4479	32.5997	32.7558
Miss America	31.5765	34.0748	34.4700	34.4419	34.9111	35.4033
Suzie	30.0295	31.4660	31.6547	31.8203	31.5208	32.1033
Foreman	0.8185	0.8708	0.8747	0.8862	0.8768	0.8924
Miss America	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

with the GNL-Means [13], BM3D [25], and 3D-KR [10]. The average PSNR and SSIM results on these three test sequences are given in Table V. As shown, the proposed method achieves better reconstruction accuracy than GNL-Means and BM3D.[3] In Figure 11, we further show the PSNR results on Foreman

[3]The PSNR results of 3D-KR are not listed, because they are not numerically available in their original papers (plotted in a figure). However, compared with their figure, our method improves over GNL-Means by a larger margin than the 3D-KR method.

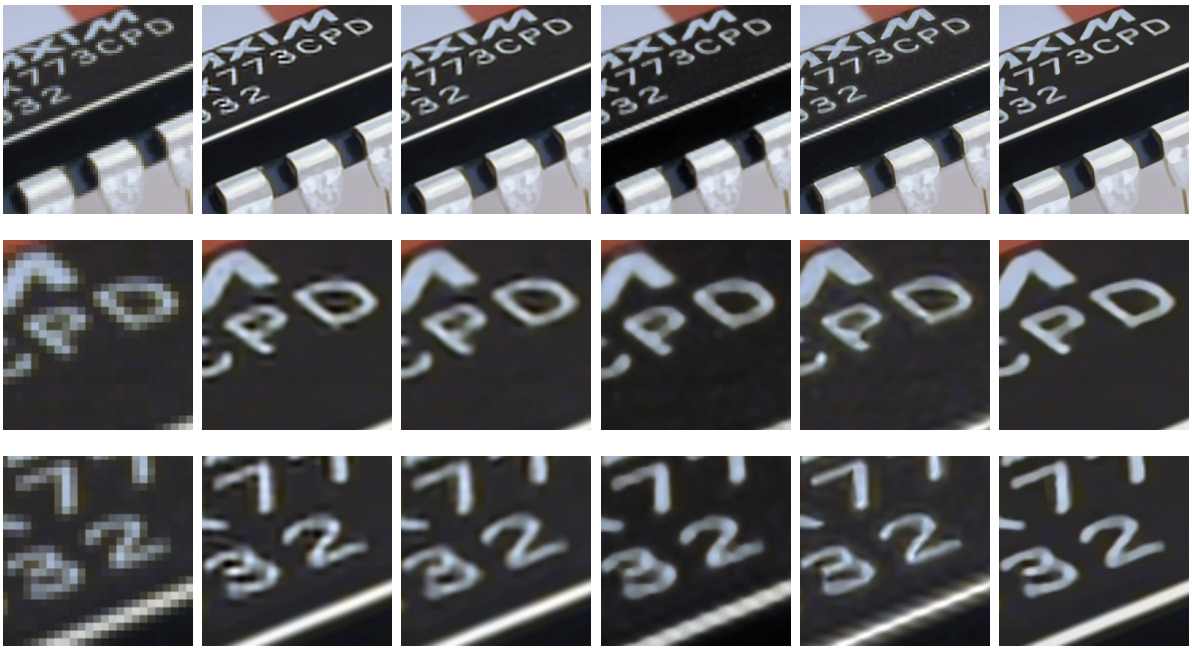

Fig. 10. **More results on single-frame SR for real color images** ($\times 4$). From left to right: NN, GNL-Means [13], KR [2], Fattal's method [35], Glasner's method [14], NL-KR. Note that Fattal's method and Glasner's method generate *jaggy* effects on the edge. Our method is free from the *jaggy* artifacts and preserves better structure.

TABLE IV

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

Sequence	NN	BI	GNL-Means [13]	BM3D [25]	Proposed
Foreman	28.8977	30.9493	34.6766	34.9	35.2041
Miss America	31.6029	34.0684	36.2508	37.5	37.8228
Suzie	30.0307	31.4702	32.9189	33.6	33.9949
Foreman	0.8413	0.8709	0.9044	–	0.9234
Miss America	0.8404	0.8928	0.8193	–	0.9346
Suzie	0.7904	0.8290	0.8428	–	0.8864

and Miss America frame by frame, compared with Bicubic and GNL-Means. The proposed method outperforms GNL-Means method by a notable margin in all frames. The SR results on Foreman and Miss America sequences are given in Figure 12 and Figure 13 respectively for visual comparison. Note that GNL-Means sometimes generates severe block artifacts (see the *Mouth* part in Figure 12 and *Eye* part in Figure 13). The 3D-KR method, on the other hand, will generate some *ghost* effects, due to overfitting of the regression and inaccurate estimation of the 3D kernel (see the *Mouth* part in Figure 12).

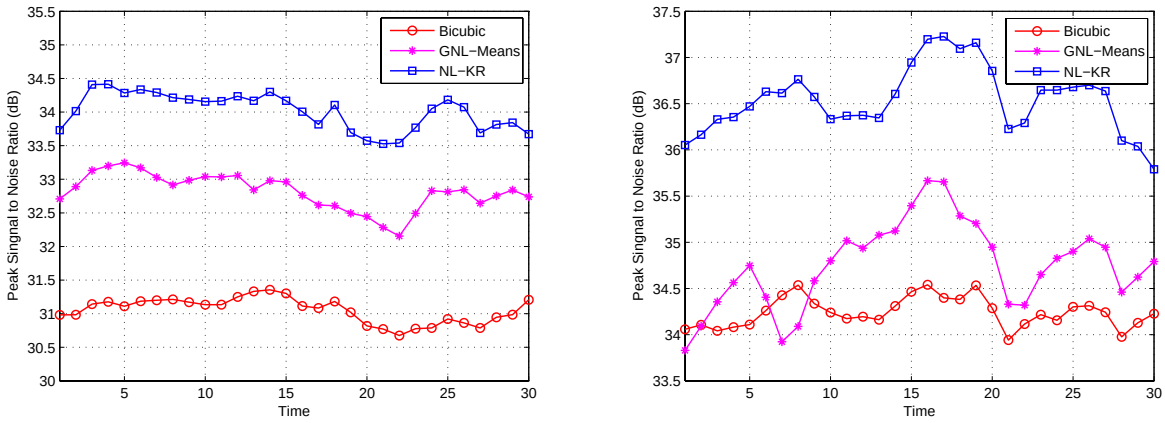

Fig. 11. **PSNR plots for Video SR**. (a) Foreman data (b) Miss America data. The proposed method outperforms other two methods in terms of PSNR in all frames.

TABLE V

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

Sequence	NN	BI	GNL-Means[13]	BM3D[25]	Proposed
Foreman	28.8444	31.0539	32.8165	33.5	34.0141
Miss America	31.6555	34.2424	35.3453	36.3	36.4377
Suzie	30.0846	31.4363	32.9725	33.0	33.0915
Foreman	0.8207	0.8720	0.9025	–	0.9120
Miss America	0.8426	0.8938	0.9136	–	0.9164
Suzie	0.7857	0.8233	0.8797	–	0.8671

Furthermore, the 3D-KR method has to employ a motion pre-compensation procedure in order for good 3D kernel estimation, while our model does not require this step. Finally, the BM3D method generates severe artifacts at edge areas, as shown in Figure 12 (second row), while our method can recover the edge structure 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 CONCLUSIONS

This paper proposes a Non-Local Kernel Regression (NL-KR) model for image and video restoration tasks, which combines the local structural regularity as well as non-local similarity explicitly to ensure a more reliable and robust estimation. The proposed method is a general model that includes many related

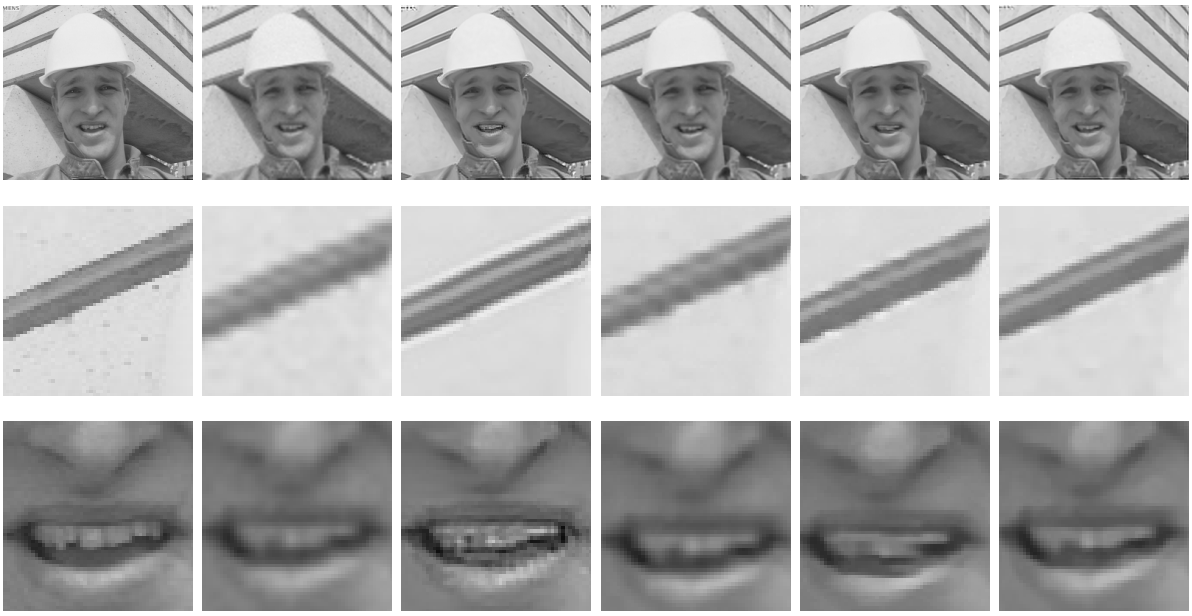

Fig. 12. **Video super-resolution for Foreman sequence** (frame 8, zoom factor 3, PSNR and SSIM in brackets). Left to right: Ground-truth, BI(30.1758, 0.8739), GNL-Means(33.2233, 0.9041), BM3D(33.45, NA), 3D-KR(33.30, NA), NL-KR(33.7589, 0.9137). GNL-Means performs well at regular-structured area but generates severe block artifacts where few similar patches can be found; BM3D suffers from *jagged* effects at edges; 3D-KR can not preserve the straight structure well due to the non-robustness of its spatial-temporal kernel and can generate *ghost image*; our method preserves both the larger structure and fine details well and is free of these artifacts.

models as special cases. We apply the proposed method to image and video denoising, deblurring and super-resolution 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 [36] searching and parallel computing respectively. Our future work would include developing fast algorithm of the proposed method and apply the proposed method to other related applications.

Acknowledgement The work is supported by the U.S. Army Research Laboratory and U.S. Army Research Office under grant number W911NF-09-1-0383. This work is also supported by National Natural Science Foundation (No.60872145, No.60903126), Cultivation Fund from Ministry of Education (No.708085), National High Technology Program (No.2009AA01Z315) and Postdoctoral (Special) Science Foundation (No.20090451397, No.201003685) of China.

Fig. 13. **Video super-resolution for Miss America sequence** (frame 8, zoom factor 3, PSNR and SSIM in brackets). Left to right: NN(32.4259, 0.8580), BI(34.5272, 0.8958), GNL-Means(34.7635,0.9132), 3D-KR(35.53, –), NL-KR(36.6509, 0.9171). GNL-Means suffers from block effects, while our method is free from artifacts and is comparable to 3D-KR in this case.

REFERENCES

[1] David Tschumperlé and R. Deriche, "Vector-valued image regularization with PDEs: A common framework for different applications," *IEEE TPAMI*, pp. 506–517, 2003.

[2] Hiroyuki Takeda, Sina Farsiu, and Peyman Milanfar, "Kernel regression for image processing and reconstruction," *IEEE TIP*, vol. 16, pp. 349–366, 2007.

[3] Michael Elad and Michal Aharon, "Image denoising via learned dictionaries and sparse representation," in *CVPR*, 2006, pp. 17–22.

[4] Jianchao Yang, John Wright, Thomas Huang, and Yi Ma, "Image super-resolution via sparse representation," *IEEE TIP*, vol. 19, no. 11, pp. 2861–2873, 2010.

[5] M. B. Wakin, D. L. Donoho, H. Choi, and R. G. Baraniuk, "The multiscale structure of non-differentiable image manifolds," in *SPIE Wavelets XI*, 2005.

[6] Gabriel Peyré, "Manifold models for signals and images," *Computer Vision and Image Understanding*, vol. 113, no. 2, pp. 249–260, 2009.

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

[8] X. Li and M. Orchard, "New edge-directed interpolation," *IEEE TIP*, vol. 10, no. 6, pp. 813–817, 2007.

[9] Xin Li, "Video processing via implicit and mixture motion models," *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 17, no. 8, pp. 953–963, Aug. 2007.

[10] Hiroyuki Takeda, Peyman Milanfar, Matan Protter, and Michael Elad, "Super-resolution without explicit subpixel motion estimation," *IEEE TIP*, 2009.

[11] Alexei Efros and Thomas Leung, "Texture synthesis by non-parametric sampling," in *In International Conference on Computer Vision*, 1999, pp. 1033–1038.

[12] Antoni Buades and Bartomeu Coll, "A non-local algorithm for image denoising," in *CVPR*, 2005.

[13] Matan Protter, Michael Elad, Hiroyuki Takeda, and Peyman Milanfar, "Generalizing the non-local-means to super-resolution reconstruction," *IEEE TIP*, pp. 36–51, 2009.

[14] Daniel Glasner, Shai Bagon, and Michal Irani, "Super-resolution from a single image," in *ICCV*, 2009.

[15] Gabriel Peyré, Sebastien Bougleux, and Laurent Cohen, "Non-local regularization of inverse problems," in *ECCV*, 2008.

[16] Haichao Zhang, Jianchao Yang, Yanning Zhang, and Thomas S. Huang, "Non-local kernel regression for image and video restoration," in *ECCV*, 2010, vol. 6313, pp. 566–579.

[17] Rafael C. Gonzalez and Richard E. Woods, *Digital Image Processing, 2/E*, Prentice Hall, 2002.

[18] Alexander Wong and Jeff Orchard, "A nonlocal-means approach to exemplar-based inpainting," in *ICIP*, 2002.

[19] Julien Mairal, Francis Bach, Jean Ponce, Guillermo Sapiro, and Andrew Zisserman, "Non-local sparse models for image restoration," in *ICCV*, 2009.

[20] Matan Protter and Michael Elad, "Super resolution with probabilistic motion estimation," *IEEE TIP*, pp. 1899–1904, 2009.

[21] David Tschumperlé, *PDE's Based Regularization of Multivalued Images and Applications*, Ph.D. thesis, 2002.

[22] Hong Chang, Dit-Yan Yeung, and Yimin Xiong, "Super-resolution through neighbor embedding," in *CVPR*, 2004.

[23] Priyam Chatterjee and Peyman Milanfar, "A generalization of non-local means via kernel regression," in *SPIE Conf. on Computational Imaging*, 2008.

[24] M. Elad and M. Aharon, "Image denoising via sparse and redundant representations over learned dictionaries," *IEEE Transactions on Image Processing*, vol. 15, no. 12, pp. 3736–3745, 2006.

[25] Aram Danielyan, Alessandro Foi, Vladimir Katkovnik, and Karen Egiazarian, "Image and video super-resolution via spatially adaptive block-matching filtering," *Int. Workshop on Local and Non-Local Approx. in Image Process*, 2008.

[26] Ramesh Neelamani, Hyeokho Choi, and Richard Baraniuk, "ForWaRD: Fourier-wavelet regularized deconvolution for ill-conditioned systems," *IEEE Transactions on Signal Processing*, vol. 52, no. 2, pp. 418–433, 2004.

[27] Haichao Zhang and Yanning Zhang, "Sparse representation based iterative incremental image deblurring," in *ICIP*, 2009.

[28] Hiroyuki Takeda, Sina Farsiu, and Peyman Milanfar, "Deblurring using regularized locally adaptive kernel regression," *IEEE Transactions on Image Processing*, vol. 17, no. 4, pp. 550–563, 2008.

[29] R. Tsai and T.S. Huang, "Multiframe image restoration and registration," *Advances in Computer Vision and Image Processing*, vol. 1, pp. 317–339, 1984.

[30] Sina Farsiu, Dirk Robinson, Michael Elad, and Peyman Milanfar, "Fast and robust multi-frame super-resolution," *IEEE Transactions on Image Processing*, vol. 13, no. 10, pp. 1327–1344, 2003.

[31] Pascal Getreuer, "http://www.math.ucla.edu/~getreuer/tvreg.html," 2009.

[32] Zhou Wang, Alan C. Bovik, Hamid R. Sheikh, Student Member, Eero P. Simoncelli, and Senior Member, "Image quality assessment: From error visibility to structural similarity," *IEEE TIP*, vol. 13, pp. 600–612, 2004.

[33] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, "Image denoising by sparse 3d transform-domain collaborative filtering," *IEEE Transactions on Image Processing*, vol. 16, no. 8, pp. 418–433, 2007.

[34] Kwang In Kim and Younghee Kwon, "Example-based learning for single-image super-resolution and jpeg artifact removal," Tech. Rep., 2008.

[35] Raanan Fattal, "Image upsampling via imposed edge statistics," in *SIGGRAPH*, 2007.

[36] Andrew Adams, Natasha Gelfand, Jennifer Dolson, and Marc Levoy, "Gaussian kd-trees for fast high-dimensional filtering," *ACM Trans. Graph*, p. 21, 2009.