Single image super-resolution via an iterative reproducing kernel Hilbert space method

Liang-Jian Deng,

School of Mathematical Sciences, University of Electronic Science and Technology of China, Chengdu, Sichuan, 611731, P. R. China

Weihong Guo, and

Department of Mathematics, Case Western Reserve University, Cleveland, OH, 44106, USA

Ting-Zhu Huang

School of Mathematical Sciences, University of Electronic Science and Technology of China, Chengdu, Sichuan, 611731, P. R. China

Abstract

Image super-resolution, a process to enhance image resolution, has important applications in satellite imaging, high definition television, medical imaging, etc. Many existing approaches use multiple low-resolution images to recover one high-resolution image. In this paper, we present an iterative scheme to solve *single* image super-resolution problems. It recovers a high quality high-resolution image from solely one low-resolution image without using a training data set. We solve the problem from image intensity function estimation perspective and assume the image contains smooth and edge components. We model the smooth components of an image using a thin-plate reproducing kernel Hilbert space (RKHS) and the edges using approximated Heaviside functions. The proposed method is applied to image patches, aiming to reduce computation and storage. Visual and quantitative comparisons with some competitive approaches show the effectiveness of the proposed method.

Index Terms

Single image super-resolution; iterative RKHS; thin-plate spline; Heaviside function

I. Introduction

IMAGE super-resolution (SR), a quite active research field currently, is a process to estimate a high-resolution (HR) image from one or multiple low-resolution (LR) images. High-resolution means more details and better visibility. Due to limitation of hardware devices and high cost, one sometimes only can collect low-resolution images. For instance, synthetic aperture radar (SAR) and satellite imaging can not get high-resolution images due to long distance and air turbulence. In medical imaging MRI, high-resolution images need more

time and cost [48], [61]. Thus, developing a more accurate and faster image super-resolution algorithm is important and has a lot of applications.

A. Literature review

In this section we review some existing super-resolution methods, some of which will be compared with the proposed method.

Many existing image super-resolution methods need multiple low-resolution images as inputs. We refer to them as multiple image super-resolution. Mathematically, there are p low-resolution images $y_i \in \mathbb{R}^m$ available, y_i is related to a high-resolution image $x \in \mathbb{R}^n$ by

$$y_i = DB_i x + n_i, 1 \leq i \leq p. \quad (1)$$

where $D \in \mathbb{R}^{m \times n}$ is a down-sampling operator and $B_i \in \mathbb{R}^{n \times n}$ is a blurring operator that might happen due to for instance out of focus, $n_i \in \mathbb{R}^m$ represents random noise [52].

This paper addresses single image super-resolution, i.e., $p = 1$ in equation (1). Compared to multiple image super-resolution, single image super-resolution is more applicable when there is only one low-resolution image available. Obviously, it is also more challenging.

Existing super-resolution methods, for both multiple images and single image, can be roughly put into several categories: interpolation-based, statistics-based, learning-based and others. This classification is by no means the best but provides an organized way for literature review. Note that, ideas of methods in different category might have overlap. For instance, some learning-based methods might also involves statistics.

Interpolation is a straightforward idea for image super-resolution. There are two popular classical interpolation methods: nearest-neighbor interpolation and bicubic interpolation. Nearest-neighbor interpolation fills in intensity at an unknown location by that of its nearest neighbor point. It often causes jaggy effect (see Figure 1(c)). Bicubic interpolation is to utilize a cubic kernel to interpolate. It tends to create blur effect (see Figure 1(b)). Recently, some state-of-the-art interpolation methods have been proposed [26], [27], [40], [45], [65], [66], [74]. In [40], for instance, it presents a new edge-directed interpolation method. It estimates local covariance coefficients from a low-resolution image, and then applies the coefficients to interpolate high-resolution image. In [74], the proposed edge-guided nonlinear interpolation bases on directional filtering and data fusion. It can preserve sharp edges and reduce ring artifacts. In [45], Mueller et al. propose an interpolation algorithm by using contourlet transform and wavelet-based linear interpolation scheme. The proposed method belongs in this category.

Maximum a Posterior (MAP) and Maximum Likelihood estimator (MLE) are popular statistics-based methods [6], [19], [20]. To preserve sharp edges, Fattal [20] utilized statistical edge dependency to relate edge features in low and high resolution images. Farsin et al. [19] proposed an alternate approach using L_1 norm minimization and a bilateral prior based robust regularization.

Learning-based approaches are a powerful tool for image super-resolution [10], [21], [23]–[25], [30], [37], [38], [41], [55], [56], [58], [60], [69], [76]. They normally start from two large training data sets, one formed of low-resolution images and the other formed of high-resolution images, and then learn a relation between low-resolution and high-resolution images. The relation is then applied to a given low-resolution image to get a high-resolution image. Learning-based methods usually can obtain high quality images but they are computationally expensive. The results might depend on the selection of training data. Additionally, they are not a completely single image super-resolution since two large data sets are required for learning. In [56], Sun et al. utilized sketch priors to extend the low vision learning approach in [25] to get clear edges, ridges and corners. Sun et al. in [55] proposed a novel profile prior of image gradient which can describe the shape and the sharpness of an image to obtain super-resolution images. Xie et al. proposed a method via an example-based strategy which divides the high-frequency patches of a low-resolution image into different classes [69]. This method can accelerate image super-resolution procedure. Fernandez-Granda and Candès used transform-invariant group-sparse regularization [21]. This method performs well for highly structured straight edges and high upscaling factors. In recent years, sparsity methods, usually associated with learning-based ideas, have been widely discussed for image super-resolution [15], [33], [70]–[73], [75]. In [71], [72], Yang et al. utilized sparse signal representation to develop a novel method for single image super-resolution. The authors first sought a sparse representation for each patch of the low-resolution image and computed corresponding coefficients, then generated the high-resolution image via the computed coefficients. Recently, Zeyde and Elad et al. [73] proposed a local sparse-land model on image patches based on the work of [71], [72], and obtained improved results.

In addition, many other image super-resolution methods also have been proposed, e.g., a frequency technique [3], pixel classification methods [1], [2], iterative back projection methods [11], [34], [39], [57], a hybrid method [14], a kernel regression method [59] and others [5], [9], [22], [53], [62].

In summary, single image super-resolution is still a challenging problem. Existing single image super-resolution methods either need training data sets and expensive computation or lead to blur or jaggy effects. The aim of this paper is to use a simple mathematical scheme to recover a high quality high-resolution image from one low-resolution image.

B. Motivation and contributions of the proposed work

In this paper, we use RKHS and Heaviside functions to study single image super-resolution with only one low-resolution image as an input. We cast the super-resolution problem as an image intensity function estimation problem. Since images contain edges and smooth components, we model them separately. We assume that the smooth components belong to a special Hilbert space called RKHS that can be spanned by a basis. We model the edges using a set of Heaviside functions. We then use intensity information of the given low-resolution image defined on a coarser grid to estimate coefficients of the basis and redundant functions, and then utilize the coefficients to generate high-resolution images at any finer grids. For

even better performance, we make the procedure iterative to recover more details, motivated by the iterative back projection method [34] and the iterative regularization method [47].

This paper has the following main contributions:

- To the best of our knowledge, this is the first work to employ RKHS method to get competitive image super-resolution results. RKHS-based methods have been considered as a powerful tool to address machine learning for a long time. In image processing, however, only limited studies have been done, e.g., image denoising [4], image segmentation [36] and image colorization [49].

- Employing Heaviside functions to recover more image details, not only getting sharp image edges, but also preserving more high-frequency details on non-edge regions.

C. Organization of this paper

The organization of this paper is as follows. In Section II, we review RKHS and splines based RKHS. We will also make two remarks in this section. In Section III, we present the proposed iterative RKHS model based on Heaviside functions and discuss the algorithms. Many visual and quantitative experiments are shown in Section IV to demonstrate the proposed method is a competitive approach for single image super-resolution. Finally, we draw conclusions in Section V.

II. Review on splines based RKHS

In this section, we review RKHS, splines based RKHS [63] and their applications in signal/ image smoothing. We will use splines based RKHS to model the smooth components of images.

A. Review on RKHS and its applications

Given a subset $\mathscr{X} \subset \mathbb{R}$ and a probability measure $\mathbb{P}$ on $\mathscr{X}$, we consider a Hilbert space $\mathcal{H}$ $\subset L^2(\mathbb{P})$, a family of functions $g \colon \mathscr{X} \to \mathbb{R}$, with $\|g\|_{L^2(\mathbb{P})} < \infty$, and an associated inner product $\langle \cdot, \cdot \rangle_{\mathcal{H}}$ under which $\mathcal{H}$ is complete. The space $\mathcal{H}$ is a reproducing kernel Hilbert space (RKHS), if there exists a symmetric function $\colon \mathscr{X} \times \mathscr{X} \to \mathbb{R}$ such that: (a) for each $x \in \mathscr{X}$, the function $(\cdot, x)$ belongs to Hilbert space $\mathcal{H}$, and (b) there exists reproducing relation $f(x) = \langle f, \mathbb{K}(\cdot, x) \rangle_{\mathscr{H}}$ for all $f \in \mathcal{H}$. Any such symmetric kernel function must be positive semidefinite (see Definition 1). Under suitable regularity conditions, Mercer's theorem [43] guarantees that the kernel has an eigen-expansion of the form

$(x, x') = \sum_{k=1}^{\infty} \lambda_k \phi_k(x) \phi_k(x')$, with $\lambda_1 \geq \lambda_2 \geq \lambda_3 \geq .. \geq 0$ being a non-negative sequence of eigenvalues, and $\{\phi_k\}_{k=1}^{\infty}$ associated eigenfunctions, taken to be orthonormal in $L^2(\mathbb{P})$.

Definition 1—(Positive Semidefinite Kernel) Let $\mathscr{X}$ be a nonempty set. The kernel $\colon \mathscr{X} \times \mathscr{X} \to \mathbb{R}$ is positive semidefinite if and only if Gram matrix $K = [\mathbb{K}(x_i, x_j)]_{N \times N}$, $x_i \in \mathscr{X}$, $i, j = 1, 2, \cdots, N$ is a positive semidefinite matrix.

Since the eigenfunctions $\{\phi_k\}_{k=1}^{\infty}$ form an orthonormal basis, any function $f \in \mathcal{H}$ has an expansion of the form $f(x) = \sum_{k=1}^{\infty} \sqrt{\lambda_k} a_k \phi_k(x)$, where $a_k = \langle f, \phi_k \rangle_{L^2(\mathbb{P})} = \int_{\mathscr{X}} f(x) \phi_k(x) d\mathbb{P}(x)$ are (generalized) Fourier coefficients. Associated with any two functions in $\mathcal{H}$, where $f = \sum_{k \geq 1} \sqrt{\lambda_k} a_k \phi_k$ and $g = \sum_{k \geq 1} \sqrt{\lambda_k} b_k \phi_k$, are two distinct inner products. The first is the usual inner product in the space $L^2(\mathbb{P})$ defined as

$\langle f, g \rangle_{L^2(\mathbb{P})} := \int_{\mathscr{X}} f(x) g(x) d\mathbb{P}(x) = \sum_{k=1}^{\infty} \lambda_k a_k b_k$. by Parseval's theorem. The second inner product, denoted $\langle f, g \rangle_{\mathcal{H}}$, defines the Hilbert space. It can be written in terms of the kernel eigenvalues and generalized Fourier coefficients as $\langle f, g \rangle_{\mathscr{H}} = \sum_{k=1}^{\infty} a_k b_k$. Using this definition, the Hilbert ball of radius 1 for $\mathcal{H}$ with eigenvalues λ_k eigenfunctions $\phi_k(\cdot)$, is

$\mathbb{B}_{\mathscr{H}}(1) = \{f \in \mathscr{H} : f(\cdot) = \sum_{k=1}^{\infty} \sqrt{\lambda_k} b_k \phi_k(\cdot) | \sum_{k=1}^{\infty} b_k^2 = \|b\|_2^2 \leq 1\}$. The class of RKHS contains many interesting classes that are widely used in practice including polynomials of degree $d((x,y) = (1 + \langle x, y \rangle)^d)$, Sobolev spaces with smoothness ν, Lipschitz, and smoothing splines. Moreover, kernel $(x, x') = \frac{1}{2} e^{-\gamma|x-x'|}$ leads to Sobolev space $\mathcal{H}_1$, i.e., a space consisted of square integrable functions whose first order derivative is square integrable. $(x, x') \propto |x - x'|$, $(x, x') \propto |x - x'|^3$ correspond to 1D piecewise linear and cubic splines respectively.

RKHS has appeared for many years, and it has been used as a powerful tool for machine learning [7], [8], [12], [13], [44], [46], [50], [54], [63]. Its application in image processing is not so common yet. In [4], Bouboulis et al. proposed an adaptive kernel method to deal with image denoising problem in the spatial domain. This method can remove many kinds of noise (e.g., Gaussian noise, impulse noise, mixed noise) and preserves image edges effectively. In addition, Kang et al. utilized RKHS method to do image segmentation [36] and image/video colorization [49].

Wahba proposed splines based RKHS for smoothing problems in [63]. It shows that the solution of an optimization problem consists of a set of polynomial splines. The proposed method is based on splines based RKHS. We thus review them in the following two subsections.

B. A 1D spline and signal smoothing

For a real-valued function

$$f \in \mathscr{G} = \{f : f \in \mathscr{C}^{m-1}[0,1], f^{(m)} \in \mathscr{L}_2[0.1]\},$$

it can be expanded at $t = 0$ by Taylor series as:

$$f(t)=\sum_{\nu=0}^{m-1}\frac{t^\nu}{\nu!}f^{(\nu)}(0)+\int_0^1\frac{(t-u)_+^{m-1}}{(m-1)!}f^{(m)}(u)du$$
$$=f_0(t)+f_1(t),\qquad\qquad(2)$$

with

$$f_0(t)=\sum_{\nu=0}^{m-1}\frac{t^\nu}{\nu!}f^{(\nu)}(0),$$

and

$$f_1(t)=\int_0^1\frac{(t-u)_+^{m-1}}{(m-1)!}f^{(m)}(u)du,$$

where $(u)_+ = u$ for $u \geq 0$ and $(u)_+ = 0$ otherwise.

Let

$$\phi_\nu(t)=\frac{t^{\nu-1}}{(\nu-1)!},\nu=1,2,\ldots,m,$$

and $\mathcal{H}_0 = \text{span}\{\phi_1, \phi_2, \ldots, \phi_m\}$ with norm $\|\phi\|^2=\sum_{\nu=0}^{m-1}\left[(D^{(\nu)}\phi)(0)\right]^2$, then $D^{(m)}(\mathcal{H}_0) = 0$. It has been proved in [63] that $\mathcal{H}_0$ is a RKHS with reproducing kernel $R^0(s,t)=\sum_{\nu=1}^{m}\phi_\nu(s)\phi_\nu(t)$. For a function $f_0 \in \mathcal{H}_0$, we can express f_0 using the basis of $\mathcal{H}_0$, i.e., $f_0(t)=\sum_{\nu=1}^{m}d_\nu\phi_\nu(t)$.

Let $\mathcal{B}_m$ be a set of functions satisfying boundary condition $f^{(\nu)}(0) = 0$, $\nu = 0, 1, 2, \cdots, m-1$ and $G_m(t,u)=\frac{(t-u)_+^{m-1}}{(m-1)!}$, then

$$f_1(t)=\int_0^1\frac{(t-u)_+^{m-1}}{(m-1)!}f^{(m)}(u)du=\int_0^1 G_m(t,u)f^{(m)}(u)du.$$

belongs to space $\mathcal{H}_1$ defined as follows:

$$\mathcal{H}_1=\{f:f \in \mathcal{B}_m, f, f', \ldots, f^{(m-1)} \text{ absolutely continuous,}$$
$$f^{(m)} \in \mathcal{L}_2\},\qquad\qquad(3)$$

where $\mathcal{H}_1$ is a Hilbert space on [0, 1] with norm $\|f\|^2 = \int_0^1 \left(f^{(m)}(t)\right)^2 dt$. $\mathcal{H}_1$ also has been proved to be a RKHS in [63] with reproducing kernel $R^1(s,t) = \int_0^1 G_m(t,u)G_m(s,u)du$. For a function $f_1 \in \mathcal{H}_1$, we can express f_1 via the basis of $\mathcal{H}_1$, denoted by $\{\xi_i\}_{i=1}^n$, so that $f_1(t) = \sum_{i=1}^n c_i \xi_i(t) = \sum_{i=1}^n c_i R^1(s_i, t)$, where $\xi_i = R^1(s_i, \cdot)$.

Due to

$$\int_0^1 \left((D^{(m)} f_0)(u)\right)^2 du = 0, \sum_{\nu=0}^{m-1} \left((D^{(\nu)} f_1)(0)\right)^2 = 0,$$

we can construct a direct sum space $\mathcal{G}_m$ by the two RKHS spaces $\mathcal{H}_0$ and $\mathcal{H}_1$, i.e., $\mathcal{G}_m = \mathcal{H}_0 \oplus \mathcal{H}_1$. $\mathcal{G}_m$ is proved as a RKHS in [63] with the following reproducing kernel

$$R(s,t) = \sum_{\nu=1}^m \phi_\nu(s)\phi_\nu(t) + \int_0^1 G_m(t,u)G_m(s,u)du, \tag{4}$$

and norm

$$\|f\|^2 = \sum_{\nu=0}^{m-1} \left[(D^{(\nu)} f)(0)\right]^2 + \int_0^1 (f^{(m)})^2(t)dt, \tag{5}$$

where $f \in \mathcal{G}_m$. As a summary, for $f \in \mathcal{G}_m$, we have $f = f_0 + f_1$, with $f_0 \in \mathcal{H}_0$, $f_1 \in \mathcal{H}_1$. It also can be written as

$$f(t) = \sum_{\nu=1}^m d_\nu \phi_\nu(t) + \sum_{i=1}^n c_i \xi_i(t), \tag{6}$$

where $t \in [0, 1]$.

let $\overrightarrow{f} = (f(t_1), f(t_1), \cdots, f(t_n))'$ be intensity values of f at $t_i \in [0, 1]$, $i = 1, 2, \cdots, n$, let

$$\overrightarrow{g} = \overrightarrow{f} + \eta, \tag{7}$$

be a noisy observation with η an additive Gaussian noise.

Let T be a $n \times m$ matrix with $T_{i,\nu} = \phi_\nu(t_i)$ and let Σ be a $n \times n$ matrix with $\Sigma_{i,j} = \langle \xi_i, \xi_j \rangle$, we have the relation $\overrightarrow{f} = Td + \sum c$ where $d = (d_1, d_2, \cdots, d_m)'$ and $c = (c_1, c_2, \cdots, c_n)'$. In [63], the following model is used to estimate $\overrightarrow{f}$ from noisy discrete measurements $\overrightarrow{g}$,

$$\min_{c,d}\frac{1}{n}\|\overrightarrow{g}-Td-\sum c\|^2+\lambda c'\sum c, \quad (8)$$

where the second term penalties nonsmoothness.

The simple model (8) has a closed-form solution:

$$c = M^{-1}(I-T(T'M^{-1}T)^{-1}T'M^{-1})\overrightarrow{g},$$
$$d = (T'M^{-1}T)^{-1}T'M^{-1}\overrightarrow{g},$$

where $M = \Sigma + n\lambda I$ with I an identity matrix. The computation burden of matrix inverse can be reduced via QR decomposition (see details in Chapter 1 of [63]).

Remark 1—Once c and d are estimated from equation (8), one can get an estimate for the signal function $f(x)$,

$$f(x)=\sum_{\nu=1}^m d_\nu \phi_\nu(x)+\sum_{i=1}^n c_i\xi_i(x)$$
$$=\sum_{\nu=1}^m d_\nu \phi_\nu(x)+\sum_{i=1}^n c_iR^1(s_i,x), \quad (9)$$

for any $x \in [0, 1]$.

Next, we will review 2D thin-plate spline which can be viewed as an extension of the mentioned 1D spline.

C. 2D thin-plate spline and image smoothing

We use 2D thin-plate spline based RKHS, introduced in [63], for image super-resolution in this paper. We thus review it.

Similar to the 1D case, let f be the intensity function of a 2D image defined on a continuous domain $E^2 = [0, 1] \times [0, 1]$. We assume f belongs in a RKHS. Let $\overrightarrow{f}=(f(t_1), f(t_1), \cdots, f(t_n))'$ be its discretization on grids $t_i = (x_i, y_i) \in [0, 1] \times [0, 1]$, $i = 1, 2, \cdots, n$, the noisy image of vector form with an additive noise η can be described by

$$\overrightarrow{g}=\overrightarrow{f}+\eta. \quad (10)$$

In [63], an optimal estimate of f for spline smoothing problems can be obtained by minimizing the following model

$$\min \frac{1}{n}\|\overrightarrow{g}-\overrightarrow{f}\|^2+\lambda J_m(f), \quad (11)$$

where m is a parameter to control the total degree of polynomial, and the penalty term is defined as follows

$$J_m(f) = \sum_{\nu=0}^{m} \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} C_m^{\nu} \left(\frac{\partial^m f}{\partial x^{\nu} \partial y^{m-\nu}} \right)^2 dx dy, \tag{12}$$

From Chapter 2 of [63], we know that the null space of the penalty function $J_m(f)$ is a $M = C_{d+m-1}^d$ dimension space spanned by the polynomials of degree no more than $m - 1$. In the experiments, we let $d = 2$ (for 2D), $m = 3$, then $M = C_{d+m-1}^d = 6$, so the null space can be spanned by the following terms: $\phi_1(x, y) = 1$, $\phi_2(x, y) = x$, $\phi_3(x, y) = y$, $\phi_4(x, y) = xy$, $\phi_5(x, y) = x^2$, $\phi_6(x, y) = y^2$. Duchon (see [18]) has proved that if there exists $\{t_i\}_{i=1}^n$ so that least squares regression on $\{\phi_{\nu}\}_{\nu=1}^M$ is unique, then the optimization model (11) has a unique solution as follows

$$f_{\lambda}(t) = \sum_{\nu=1}^{M} d_{\nu} \phi_{\nu}(t) + \sum_{i=1}^{n} c_i E_m(t, t_i), \tag{13}$$

where $E_m(t, t_i)$ is a Green's function for the m-iterated Laplacian defined as:

$$E_m(s, t) = E_m(|s - t|) = \theta_{m,d} |s - t|^{2m-d} \ln |s - t|,$$

where $\theta_{m,d} = \dfrac{(-1)^{d/2+m+1}}{2^{2m-1} \pi^{d/2} (m-1)! (m-d/2)!}$, especially, $E_m(t, t_i)$ plays the same role with $\xi_i(t)$ in 1D case.

Similar with equation (8), model (11) can be rewritten as:

$$\min \frac{1}{n} \| \overrightarrow{g} - (Td + Kc) \|^2 + \lambda c' K c, \tag{14}$$

where T is a $n \times M$ matrix with $T_{i,\nu} = \phi_{\nu}(t_i)$ and K is a $n \times n$ matrix with $K_{i,j} = E_m(t_i, t_j)$. This model also has a similar closed-form solution with 1D case:

$d = (T'W^{-1}T)^{-1} T'W^{-1} \overrightarrow{g}$, $c = W^{-1}(I - T(T'W^{-1}T)^{-1} T'W^{-1}) \overrightarrow{g}$ where $W = K + n\lambda I$. Additionally, a more economical version that utilizing QR decomposition also has been provided to compute the coefficients c and d (see details in [63]). Moreover, more information about the thin-plate spline can also be found in [16]–[18], [42], [51], [64].

Remark 2—Once we have computed coefficients c and d, the underlying function f on the continuous domain E^2 can be estimated as

$$f(w) = \sum_{\nu=1}^{M} d_\nu \phi_\nu(w) + \sum_{i=1}^{n} c_i E_m(t_i, w), \quad (15)$$

for any $w = (x, y)' \in E^2$. One thus can get an estimate of $f(w)$ at any $w \in [0, 1] \times [0, 1]$. This is very powerful and makes image super-resolution possible.

III. The proposed iterative method

Let f represent intensity function of an image defined on a continuous domain. Without loss of generality, we assume the domain is $E^2 = [0, 1] \times [0, 1]$. Let H, L be a high-resolution and a low-resolution discretization of f, respectively. For notation simplicity, we interchangeably use H, L to represent their matrix and vector representations. H and L are usually formulated by $L = DBH + \varepsilon$ as described in equation (1), with D, B a down-sampling and a blurring operator, respectively, ε some random noise or 0 for noise-free case. We note that the high-resolution image $H \in \mathbb{R}^{U \times V}$ can be obtained by $H_i = f(t_i^h)$ with $t_i^h = (x_i, y_i)$,

$x_i \in \{0, \frac{1}{U-1}, \frac{2}{U-1}, \cdots, 1\}$, $y_i \in \{0, \frac{1}{V-1}, \frac{2}{V-1}, \cdots, 1\}$ on a finer grid. Low-resolution image $L \in \mathbb{R}^{Q \times S}$ is gotten by the discretization formula $L_i = f(t_i^l)$ with $t_i^l = (x_i, y_i)$,

$x_i \in \{0, \frac{1}{Q-1}, \frac{2}{Q-1}, \cdots, 1\}$, $y_i \in \{0, \frac{1}{S-1}, \frac{2}{S-1}, \cdots, 1\}$ on a coarser grid. In particular, Q, S are smaller than U, V, respectively. Actually, $T^l \in \mathbb{R}^{n \times M}$, $K^l \in \mathbb{R}^{n \times n}$ and $T^h \in \mathbb{R}^{N \times M}$, $K^h \in \mathbb{R}^{N \times n}$ are the T, K matrices and $\tilde{T}$, $\tilde{K}$ matrices in Section II-C, respectively, where $n = Q \cdot S$, $N = U \cdot V$ and M is the dimension of the null space of the penalty term (see details also in Section II-C). Motivated by the smoothing model (14), c, d can be solved using the following model

$$\min \frac{1}{n} \| L - DB(T^h d + K^h c) \|^2 + \lambda c' K^l c, \quad (16)$$

where $H = T^h d + K^h c$.

However, model (16) is for image smoothing. Super-resolution results via this model may smooth out image edges. In this work, we employ Heaviside functions to recover more images details such as edges.

A. Heaviside function

Heaviside function, or Heaviside step function (see Figure 2(a)), is defined as follows

$$\phi(x) = \begin{cases} 0, & x < 0, \\ 1, & x \geq 0. \end{cases} \quad (17)$$

The Heaviside function is singular at $x = 0$ and describes a jump at $x = 0$ perfectly. We usually use its smooth approximation for practical problems. In our work, we use the following approximated Heaviside function (AHF),

$$\psi(x) = \frac{1}{2} + \frac{1}{\pi}\arctan\left(\frac{x}{\xi}\right), \quad (18)$$

which approximates to $\phi(x)$ when $\xi \to 0$ and $\xi \in \mathcal{R}$ actually controls the smoothness. The smaller ξ the sharper jump (see Figure 2(b)).

The AHF $\psi(\cdot)$ is a 1D function. Its variation $\psi(\mathbf{v}_i\mathbf{x} + c_i)$ is however a 2D function when $\mathbf{x} \in \mathcal{R}^2$. If we let $\mathbf{v}_i = (\cos\theta_i, \sin\theta_i)$, $\psi(\mathbf{v}_i\mathbf{x} + c_i)$ can actually describe an edge with orientation θ_i located at a position specified by c_i. In Figure 3, we show some examples of $\psi((\cos\theta_i, \sin\theta_i)\cdot\mathbf{z} + c_i)$. One can see that as θ_i, c_i vary, we get edges of various orientation at different locations. Confirmed by the following theoretical foundation, we model edges in 2D images using linear combination of this type of function.

Theorem 1—(see [35]) For any positive integers m, d and any $p \in [1, \infty)$,

$span_m H_d = \{\sum_{i=1}^{m} \omega_i \psi(\mathbf{v}_i \cdot \mathbf{x} + c_i)\}$, with $\omega_i \in \mathcal{R}$ $\mathbf{v}_i \in \mathcal{R}^d$ and $c_i \in \mathcal{R}$, is approximately a compact subset of $(\mathcal{L}_p([0, 1]^d), \|\cdot\|_p)$.

2D images are defined in $\mathcal{R}^2$, i.e., $d = 2$. Based on the above theorem, we model edges in 2D images using the following:

$$g(\mathbf{z}) = \sum_{j=1}^{m} \omega_j \psi\left((\cos\theta_j, \sin\theta_j) \cdot \mathbf{z} + \mathbf{c_j}\right), \quad (19)$$

where small $\xi = 10^{-4}$ is used in ψ, and

$$\theta_j \in \{0, \pi/12, 2\pi/12, 3\pi/12, \ldots, 23/12\pi\},$$

while $c_j \in \{0, \frac{1}{n-1}, \frac{2}{n-1}, \ldots, 1\}$ where n is the number of all pixels of low-resolution image, $m = kn$ where k is the number of orientations $\{\theta_j\}$.

Actually, equation (19) can be written as $g = \Psi\omega$ where $\Psi \in \mathcal{R}^{n \times m}$, $g \in \mathcal{R}^n$, $\omega \in \mathcal{R}^m$.

B. The proposed iterative method based on RKHS and Heaviside functions

In this work, we assume the underlying image intensity function f is the sum of smooth components and edges, which are modeled using splines based RKHS and Heaviside functions, i.e., $f = Td + Kc + \Psi\beta$. Since Ψ contains a pretty exhaustive list of functions while edges are pretty sparse in images, it is thus reasonable to expect β to be sparse. The final proposed model is as follows

$$\min \frac{1}{n}\|L - DB(T^h d + K^h c + \Psi^h \beta)\|^2 + \lambda c' K^l c + \alpha \|\beta\|_1, \quad (20)$$

where $H = T^h d + K^h c + \Psi^h \beta$ and ℓ sparsity is enforced for β. For blur free case, $B = I$, an identity matrix, $DB(T^h d + K^h c + \Psi^h \beta)$ is considered as $T^l d + K^l c + \Psi^l \beta$. Since $\|\beta\|_1$ is not differentiable, we make a variable substitution and solve the following equivalent problem:

$$\min \frac{1}{n}\|L - (T^l d + K^l c + \Psi^l \beta)\|^2 + \lambda c' K^l c + \alpha \|u\|_1, s.t., u = \beta. \quad (21)$$

using alternating direction method of multipliers (ADMM) that is a very popular method for solving L_1 problem [29], [31], [67]. In particular, the convergence of ADMM method is guaranteed by many works, e.g., [28], [32]. The augmented Lagrangian of problem (21) is as follows

$$\mathcal{L}(c, d, \beta, u) = \frac{1}{n}\|L - (T^l d + K^l c + \Psi^l \beta)\|^2 + \lambda c' K^l c \\ + \alpha \|u\|_1 + \frac{\rho}{2}\|u - \beta + b\|^2, \quad (22)$$

where $a, \rho \in \mathcal{R}$ are regularization parameters, b is a Lagrangian multiplier.

The energy functional in (22) is separable with respect to (c, d, β) and u. We can thus focus on the two subproblems:

$$(c, d, \beta) - \text{Subproblem:} \\ \min_{(c,d,\beta)} \frac{1}{n}\|L - (T^l d + K^l c + \Psi^l \beta)\|^2 \\ + \lambda c' K^l c + \frac{\rho}{2}\|u - \beta + b\|^2, \quad (23)$$

$$u - \text{subproblem:} \min_u \alpha \|u\|_1 + \frac{\rho}{2}\|u - \beta + b\|^2. \quad (24)$$

The u-subproblem (24) has a closed form solution and is calculated for each u_i (see [67]) as

$$u_i = shrink(\beta_i - b_i, \frac{\alpha}{\rho}), \quad (25)$$

where $shrink(a, b) = sign(a)\max(|a-b|, 0)$ and $0.(0/0) = 0$ is assumed.

We employ least squares method to solve the (c, d, β)-subproblem (23). The normal equation reads as

$$\begin{pmatrix} K^{l'}K^{l}+n\lambda K^{l} & K^{l'}T^{l} & K^{l'}\Psi^{l} \\ T^{l'}K^{l} & T^{l'}T^{l} & T^{l'}\Psi^{l} \\ \Psi^{l'}K^{l} & \Psi^{l'}T^{l} & \Psi^{l'}\Psi^{l}+\frac{n\rho}{2}I \end{pmatrix} \begin{pmatrix} c \\ d \\ \beta \end{pmatrix}$$
$$= \begin{pmatrix} K^{l'}L \\ T^{l'}L \\ \Psi^{l'}L+\frac{n\rho}{2}(u+b) \end{pmatrix}, \tag{26}$$

Equation (26) can be rewritten as the following three equations,

$$(K^{l'}K^{l}+n\lambda K^{l})c+K^{l'}T^{l}d+K^{l'}\Psi^{l}\beta=K^{l'}L, \tag{27}$$

$$T^{l'}K^{l}c+T^{l'}T^{l}d+T^{l'}\Psi^{l}\beta=T^{l'}L, \tag{28}$$

$$\Psi^{l'}K^{l}c+\Psi^{l'}T^{l}d+(\Psi^{l'}\Psi^{l}+\frac{n\rho}{2}I)\beta=\Psi^{l'}L+\frac{n\rho}{2}(u+b). \tag{29}$$

We can solve for β from equation (29) in terms of c, d:

$$\beta=(\Psi^{l'}\Psi^{l}+\frac{n\rho}{2}I)^{-1}(\Psi^{l'}L+\frac{n\rho}{2}(u+b)-\Psi^{l'}K^{l}c-\Psi^{l'}T^{l}d). \tag{30}$$

We then substitute equation (30) into equation (27) and equation (28) and obtain

$$c=(A_{1}-A_{3}A_{4}^{-1}A_{2})^{-1}(e_{1}-A_{3}A_{4}^{-1}e_{2}),$$
$$d=A_{4}^{-1}(e_{2}-A_{2}c),$$
$$\beta=(\Psi^{l'}\Psi^{l}+\frac{n\rho}{2}I)^{-1}(\Psi^{l'}L+\frac{n\rho}{2}(u+b)-\Psi^{l'}K^{l}c-\Psi^{l'}T^{l}d). \tag{31}$$

where $A_{1}=(K^{l'}K^{l}+n\lambda K^{l})-K^{l'}\Psi^{l}(\Psi^{l'}\Psi^{l}+\frac{n\rho}{2}I)^{-1}\Psi^{l'}K^{l}$,

$A_{2}=T^{l'}K^{l}-T^{l'}\Psi^{l}(\Psi^{l'}\Psi^{l}+\frac{n\rho}{2}I)^{-1}\Psi^{l'}K^{l}$, $A_{3}=K^{l'}T^{l}-K^{l'}\Psi^{l}(\Psi^{l'}\Psi^{l}+\frac{n\rho}{2}I)^{-1}\Psi^{l'}T^{l}$,

$A_{4}=T^{l'}T^{l}-T^{l'}\Psi^{l}(\Psi^{l'}\Psi^{l}+\frac{n\rho}{2}I)^{-1}\Psi^{l'}T^{l}$,

$e_{1}=K^{l'}L-K^{l'}\Psi^{l}(\Psi^{l'}\Psi^{l}+\frac{n\rho}{2}I)^{-1}(\Psi^{l'}L+\frac{n\rho}{2}(u+b))$,

$e_{2}=T^{l'}L-T^{l'}\Psi^{l}(\Psi^{l'}\Psi^{l}+\frac{n\rho}{2}I)^{-1}(\Psi^{l'}L+\frac{n\rho}{2}(u+b))$. Equation (31) looks complicated and involves some matrix inversions, but we only compute it once in the algorithm and the

matrix inversions are not ill-conditioned with proper λ and ρ. If we apply the algorithm to image patches (see details in the end of this section), the computation is very cheap.

The following algorithm is the corresponding ADMM scheme:

Note that the convergence of Algorithm 1 is guaranteed by the following theorem that its proof can be found in [28].

Theorem 2—For any $\gamma \in (0, (\sqrt{5} + 1)/2)$, the sequence $\{(c^{(j)}, d^{(j)}, \beta^{(j)})\}$ obtained by Algorithm 1 converges to the solution of problem (20) for any initial points $u^{(0)}$ and $b^{(0)}$.

In particular, we set $\gamma = 1$, $\beta^{(0)} = b^{(0)} = 0$ in our work, the convergence of Algorithm 1 thus can be guaranteed.

Although model (20) can pick up more image details, it can not completely overcome blur effect along edges of high-resolution image. Due to imperfect reconstruction from the model, we observe residual edges in difference image $L - DBH^{(1)}$ where $H^{(1)}$ is the computed high-resolution image by model (20). Inspired by the iterative back projection method [34] and the iterative regularization method [47], we consider the difference $L - DBH^{(1)}$ as a new low-resolution input L, and recompute model (20) to get a residual high-resolution image $H^{(2)}$. We repeat this process until the residual is small enough. The sum of the high-resolution image $H^{(1)}$ and its residual high-resolution images is the resulted super-resolution image H. The strategy can recover more image details (see Figure 4). In our experiments, it is enough to iterate the process ten times. Algorithm 2 summarizes the proposed iterative RKHS algorithm for single image super-resolution. This algorithm can work for general D, B though we mainly tested it with bicubic down-sampling and blur-free in the experiments.

For Algorithm 2, note that although we introduce some parameters in the super-resolution algorithm, these parameters, are all not sensitive and easy to select (see the parameters remark in Section IV). The solution of step 3a is obtained by Algorithm 1. Down-sampling operators D, associating with step 3a and step 3c in Algorithm 2, are done by bicubic interpolation (In Matlab function: "imresize").

Algorithm 2 can be applied on the whole image or patch by patch. In our numerical experiments, we apply the algorithm to image patches to reduce computation time and storage. We set patch size to be 6×6 with overlaps. Intensity at the boundary is estimated by bicubic interpolation.

In what follows, we compare the proposed approach with some competitive methods.

IV. Numerical experiments

In this section, we mainly compare the proposed approach with some state-of-the-art super-resolution methods: bicubic interpolation, a fast upsampling method ("08'TOG" [53]), a learning-based method ("10'TIP" [71]). In addition, the proposed method actually can be viewed as an interpolation-based approach. Thus it is necessary to compare the proposed

method with some state-of-the-art interpolation methods, e.g., two contour stencils based interpolations ("11'IPOL" [27], "11'SIAM" [26]) and an interpolation and reconstruction based method ("14'TIP" [65]). Furthermore, we also compare the proposed method with a kernel regression method ("07'TIP" [59]), multiscale geometric method ("07'SPIE" [45]).

We use two kinds of test images. One is low-resolution images without high-resolution ground-truth (see Section IV-A). The other is simulated low-resolution images from known high-resolution images (see Section IV-B). In the later case, one has high-resolution ground-truth available for quantitative comparisons. For fair comparison, we set $B = I$ in our experiments because some of the methods compared do not involve deblurring process. All experiments are done in MATLAB(R2010a) on a laptop of 3.25Gb RAM and Intel(R) Core(TM) i3-2370M CPU: @2.40 GHz, 2.40 GHz.

The proposed Algorithm 2 is for gray-scale images. For color images such as RGB, there is redundancy in channels, we first transform it to "YCbCr" color space[1] where "Y" represents luminance component, "Cb" and "Cr" represent blue-difference and red-difference components that are less redundant. "Y" is essentially a grayscale copy of the color image and carries most of the high resolution details of the color image. This color space is very popular in image/video processing. Because humans are more sensitive to luminance changes, the proposed algorithm is only applied to the illuminance channel and bicubic interpolation is applied to the color layers (Cb, Cr). The upscaled images in YCbCr space is transformed back to the original color space for visualization/analysis. Color image results are better visualized in the original pdf file.

We employ root-mean-square error (RMSE) for quantitative comparisons, and the RMSE index is used in some super-resolution works, e.g., "10'TIP" [71]. Furthermore, a popular index Peak Signal-Noise Ration (PSNR) is utilized to estimate the performance of different methods. In particular, we compute PNSR only on the luminance channel "Y" in the experiments. In addition, we also employ the structural similarity (SSIM) index[2] [68] to compare different methods.

A remark on parameter selection

The related parameters in Algorithm 1 and Algorithm 2 are easy to select. We set $\lambda = 10^{-11}$, $a = 10^{-4}$, $\rho = 10^{-5}$. The maximum iteration τ is 3. For simplicity, we only do 10 iterations for Algorithm 1. In addition, we set $M = 6$ so that $\phi_1(t) = 1$, $\phi_2(t) = x$, $\phi_3(t) = y$, $\phi_4(t) = xy$, $\phi_5(t) = x^2$, $\phi_6(t) = y^2$ (see details in Section II-C). Note that the proposed method includes many parameters, e.g., λ, ρ, patch size, etc. However, they are easy to select because the proposed method that can be viewed as an interpolation approach is not sensitive to the selection of parameters. Actually, choosing suitable parameters is always a difficulty to many image algorithms. Tuning empirically is a popular way to determine parameters. In our work, we obtain the parameters by tuning empirically.

---

[1]http://en.wikipedia.org/wiki/YCbCr
[2]https://ece.uwaterloo.ca/~z70wang/research/ssim/

A. Results on low-resolution images without ground-truth

In this section, experiments are based on natural images without ground-truth, thus quantitative comparisons (e.g., RMSE) are not available.

In Figure 5 and Figure 6, we compare the proposed SR-RKHS method with classical bicubic interpolation, "07'TIP" [59], "08'TOG" [53], "10'TIP" [71], "11'IPOL" [27], "11'SIAM" [26] and "14'TIP" [65]. The upscaling factors are all 3. From the figures, the results of bicubic interpolation, "07'TIP" and "08'TOG" show blur effect for the whole image. The results of "10'TIP" and "14'TIP" preserve sharp edges well, however, they smooth out image details on non-edge regions, e.g., freckles on the skin (see close-ups in Figure 6). The two contour interpolation methods "11'IPOL" and "11'SIAM" keep image edges and details well, but the results contain some artificial contours near true edges. The proposed method performs well, not only on edges but also for fine details/textures away from edges.

B. Results on low-resolution images simulated from known ground-truth images

To provide quantitative comparisons in terms of RMSE, PSNR and SSIM, we start from some high-resolution images, treat them as ground-truth and simulate low-resolution images by bicubic interpolation.

In this section, we mainly compare the proposed method with several state-of-the-art methods: bicubic interpolation, "08'TOG" [53], "10'TIP" [71], "11'IPOL" [27] and "14'TIP" [65]. In Figures 7–9, upscaled high-resolution images by bicubic interpolation show blur effect. Although we can get sharp edges via "08'TOG" [53], it flattens details on non-edge regions. The method "11'IPOL" [27] recovers image details well, but introduces some artificial contours near true edges. For instance, for the "baboon" example in Figure 8, it has many artificial contours near the true edges (see the closeup). "14'TIP" [65] preserves sharp image edges but smooths out image intensity not on edges. The method "10'TIP" [71] obtains competitive visual results, however, it generates worse quantitative results than the proposed method (see Table I). In addition, the results of "07'SPIE" [45] and "11'SIAM" [26] in Figure 9 also perform worse than the proposed method. The proposed method not only preserves sharp edges but also keeps high-frequency details well on non-edge regions. Furthermore, the proposed method also gets the best RMSE, PSNR and SSIM for almost all examples.

In Figure 10 and Table I, we find that the proposed method gets better quantitative and visual results. The results of bicubic interpolation and "08'TOG" show significant blur effect. The method "11'IPOL" also obtains excellent visual results, but the visual results show obvious artificial contours. The method "14'TIP" gets the sharpest image edges, but it smoothes out image details on non-edge regions. In addition, The method "10'TIP" obtains similarly visual results with the proposed method, but the proposed method has lower RMSE, larger PSNR and SSIM. In Figure 11, the proposed method performs best, especially for image details, e.g., hair of lion. The learning-based method "10'TIP" [71] obtain excellent visual and quantitative results, it however needs extra training data to generate dictionary. We also give corresponding error maps in Figure 12. Furthermore, we can find more quantitative comparisons in Table I. It demonstrates that the proposed method gets

better quantitative performance than other methods for almost all examples. In particular, instead of RKHS and Heaviside functions, one can use wavelet basis or frames in our framework. We haven't got time to compare the performance.

Computation issue—We present the computation comparisons in Table II. From the table, we find that bicubic interpolation is the fastest. However, we have to note that bicubic interpolation is optimized in MATLAB, "08'TOG" is optimized by an executable software[3], "11'IPOL"[4] and "14'TIP"[5] are speeded up via C language and Cmex, respectively. Only "10'TIP"[6] and the proposed method are based on MATLAB codes that are not optimized. In particular, computation time with respect to the change of upscaling factor and image size is presented in Figure 13. One can see that it is acceptable to employ our method for image super-resolution. The computation time is based on non-optimized Matlab code. It has a lot of room to speed up the code. For instance, the code contains a lot of loops that can be significantly sped up using Cmex.

The relation between model (20) and model (20) combined with iterative strategy—Equation (20) is the proposed model in the work. In particular, we employ an iterative strategy for the proposed model to recover more image details. Thus it is necessary to illustrate the relation between model (20) and model (20) combined with our iterative strategy. Actually, there is no significantly visual difference between the two methods, especially in image details and edges (see the almost dark error map in Figure 14(c)). However, it is easy to know that the proposed model (20) combined with the iterative strategy performs lower RMSE comparing with the proposed model (20). In addition, the iterative strategy results in more computation obviously.

V. Conclusions

Given a low-resolution image, the super-resolution problem was casted as an image intensity function estimation problem. Because images mainly contain smooth components and edges, we assumed smooth components belong to 2D thin-plate spline based RKHS and edges can be represented by approximated Heaviside functions. The coefficients of the redundant basis were computed using the low-resolution image. We then applied the coefficients to generate high-resolution images. To recover sharp high-resolution images, we proposed an iterative scheme to preserve more image details. In addition, we applied the proposed method to image patches to reduce computation and storage significantly. Many experiments showed that the proposed approach outperformed the state-of-the-art methods, both visually and quantitatively.

Acknowledgments

Biographies

Liang-Jian Deng received B.S. degree from the School of Mathematical Sciences, University of Electronic Science and Technology of China(UESTC), Chengdu, China, in 2010. He is currently pursuing the Ph.D. degree with School of Mathematical Sciences of UESTC. His current research interest is image processing, including image super-resolution, deblurring&denoising, inpainting and dehazing.

Weihong Guo received the B.S. degree in Computational Math from Minzu University of China in 1999, the M.S. degree in Statistics and the Ph.D. degree in Applied Math, both from University of Florida and both in 2007. She was a Math Assistant Professor at the University of Alabama 2007–2009 and is now an Applied Math Associate Professor at Case Western Reserve University, OH. Her research interests include variational image reconstruction, image super-resolution and image segmentation.

Ting-Zhu Huang is a professor at the School of Mathematical Sciences, University of Electronic Science and Technology of China. His research interests include numerical linear algebra and scientific computation with applications in electromagnetics, modeling and algorithms for image processing, etc. He has published over 100 papers in international journals, including SIAM J. Sci. Comput., SIAM J. Matrix Anal. Appl., IMA J. Numerical

Anal., J. Comput. Phys., Computer Phys. Comm., Numerical Lin. Alg. Appl., Automatica, IEEE Trans. Antennas J. Matrix Anal. Appl., IMA J. Numerical Anal., J. Comput. Phys., Computer Phys. Comm., Numerical Lin. Alg. Appl., Automatica, IEEE Trans. Antenna and propagation, IEEE Trans. Geoscience and Remote Sensing, Information Sciences, J. Optical Society of America A, Computing, Lin. Alg. Appl., Appl. Math. Letters, Comput. Math. Appl., Appl. Math. Modelling, J. Franklin Institute, J. Comput. Appl. Math., Comm. Nonlin. Sci. Numer. Simul., etc. He received the Science and Technology Progress Award of Sichuan Province, Chinese Information Ministry for several times. Dr. Huang has been served in the editorial board of several international journals.

References

1. Atkins CB, Bouman CA, Allebach JP. Tree-based resolution synthesis. International Conference on Image Processing (ICIP). 1999:405–410.

2. Atkins CB, Bouman CA, Allebach JP. Optimal image scaling using pixel classification. International Conference on Image Processing (ICIP). 2001:864–867.

3. Borman S, Stevenson RL. Super-resolution from image sequences - a review. Midwest Symposium on Circuits and Systems. 1998:374–378.

4. Bouboulis P, Slavakis K, Theodoridis S. Adaptive kernel-based image denoising employing semi-parametric regularization. IEEE Transactions on Image Processing. 2010; 19:1465–1479. [PubMed: 20236901]

5. Candès E, Fernandez-Granda C. Towards a mathematical theory of super-resolution. To appear in Communications on Pure and Applied Mathematics.

6. Capel D, Zisserman A. Super-resolution enhancement of text image sequences. International Conference on Pattern Recognition (ICPR). 2000; 1:600–605.

7. Caponnetto A, Pontil M, Micchelli C, Ying Y. Universal multitask kernels. Journal of Machine Learning Research (JMLR). 2008; 9:1615–1646.

8. Carmeli C, De Vito E, Toigo A. Vector valued reproducing kernel Hilbert spaces of integrable functions and Mercer theorem. Analysis and Applications. 2006; 4:377–408.

9. Chambolle A, Pock T. A First-Order Primal-Dual Algorithm for Convex Problems with Applications to Imaging. Journal of Mathematical Imaging and Vision. 2011; 40:120–145.

10. Chang H, Yeung D, Xiong Y. Super-Resolution through neighbor embedding. Computer Vision and Pattern Recognition (CVPR). 2004; 1

11. Chatterjee P, Mukherjee S, Chaudhuri S, Seetharaman G. Application of Papoulis-Gerchberg method in image super-resolution and inpainting. The Computer Journal. 2007; 52:80–89.

12. Coifman RR, Lafon S. Geometric harmonics: a novel tool for multiscale out-of-sample extension of empirical functions. Applied and Computational Harmonic Analysis. 2006; 21:31–52.

13. Cucker F, Smale S. On the mathematical foundations of learning. Bulletin of the American Mathematical Society. 2002; 39:1–49.

14. Daniel G, Bagon S, Irani M. Super-Resolution from a single image. ICCV. 2009:349–356.

15. Dong W, Shi G, Zhang L, Wu X. Superresolution with nonlocal regularized sparse representation. Proceeding of SPIE. 2010

16. Duchon J. Fonctions splines et vecteurs aleatoires. Tech Report 213, Seminaired Analyse Numerique, Universite Scientifique et Medicale, Grenoble. 1975

17. Duchon J. Fonctions-spline et esperances conditionnelles de champs gaussiens. Ann Sci Univ Clermont Ferrand II Math. 1976:19–27.

18. Duchon J. Splines minimizing rotation-invariant semi-norms in Sobolev spaces. Constructive Theory of Functions of Several Variables. 1977:85–100.

19. Farsiu S, Robinson MD, Elad M, Milanfar P. Fast and robust multiframe super resolution. IEEE transactions on image processing. 2004; 13:1327–1344. [PubMed: 15462143]

20. Fattal R. Image upsampling via imposed edge statistics. ACM Transactions on Graphics. 2007; 26

21. Fernandez-Granda C, Candès E. Super-resolution via transform-invariant group-sparse regularization. ICCV. 2013

22. Freedman G, Fattal R. Image and video upscaling from local self-examples. ACM Trans on Graphics (TOG). 2011; 30

23. Freeman WT, Jones TR, Pasztor EC. Example-based super-resolution. IEEE Computer Graphics and Applications. 2002; 22:56–65.

24. Freeman WT, Pasztor EC. Markov networks for super-resolution. Proceedings of 34th Annual Conference on Information Sciences and Systems. 2000

25. Freeman WT, Pasztor EC, Carmichael OT. Learning low-level vision. International Journal of Computer Vision. 2000; 40:25–47.

26. Getreuer P. Contour stencils: Total Variation along curves for adaptive image interpolation. SIAM Journal on Imaging Sciences. 2011; 4:954–979.

27. Getreuer P. Image interpolation with contour stencils. Image Processing On Line. 2011; 1

28. Glowinski, R., Tallec, PL., Sheikh, HR., Simoncelli, EP. Augmented Lagrangian and Operator Splitting Methods in Nonlinear Mechanics. Texas: 1989.

29. Goldstein T, Osher S. The split bregman method for l1-regularized problems. SIAM Journal on Imaging Sciences. 2009; 2:323–343.

30. Gur E, Zalevsky Z. Single-Image digital super-resolution a revised Gerchberg-Papoulis algorithm. IAENG International Journal of Computer Science. 2007; 34:251–255.

31. He B, Tao M, Yuan X. Alternating Direction Method with Gaussian Back Substitution for Separable Convex Programming. SIAM Journal on Optimization. 2012; 22:313–340.

32. He B, Yang H. Some convergence properties of a method of multipliers for linearly constrained monotone variational inequalities. Operations Research Letters. 1998; 23:151–161.

33. He L, Qi H, Zaretzki R. Beta process joint dictionary learning for coupled feature spaces with application to single image super-resolution. CVPR. 2013:345–352.

34. Irani M, Peleg S. Super resolution from image sequence. Proceedings of 10th International Conference on Pattern Recognition (ICPR). 1990:115–120.

35. Kainen PC, Kurková V, Vogt A. Best approximation by linear combinations of characteristic functions of half-space. Journal of Approximation Theory. 2003; 122:151–159.

36. Kang SH, Shafei B, Steidl G. Supervised and Transductive Multi-Class Segmentation Using p-Laplacians and RKHS methods. Preprint at uni-klde. 2012

37. Kim C, Choi K, Hwang K, Ra JB. Learning-based super-resolution using a multi-resolution wavelet approach. Iternational workshop on Advance Image Technology (IWAIT). 2009

38. Kim C, Choi K, Ra JB. Improvement on learning-based super-resolution by adopting residual information and patch reliability. IEEE International Conference on Image Processing (ICIP). 2009:1197–1200.

39. Komatsu K, Igarashi T, Saito T. Very high resolution imaging scheme with multiple different-aperture cameras. Signal Processing: Image Communication. 1993; 5:511–526.

40. Li X, Orchard M. New Edge-Directed Interpolation. IEEE Trans Image Processing. 2001; 10:1521–1527.

41. Liyakathunisa, Ananthashayana VK. Super resolution blind reconstruction of low resolution images using wavelets based fusion. International Journal of Computer and Information Engineering. 2008; 2:106–110.

42. Meinguet J. Multivariate interpolation at arbitrary points made simple. Journal of Applied Mathematics and Physics (ZAMP). 1979; 30:292–304.

43. Mercer J. Functions of positive and negative type, and their connection with the theory of integral equations. Philosophical transactions of the royal society of London Series A, containing papers of a mathematical or physical character. 1909; 209:415–446.

44. Micchelli CA, Pontil M. On leaning vector-valued functions. Neural Computation. 2005; 17:177–204. [PubMed: 15563752]

45. Mueller N, Lu Y, Do M. Image Interpolation Using Multiscale Geometric Representations. SPIE proceedings. 2007

46. Nosedal-Sanchez A, Storlie CB, Lee TCM, Christensen R. Reproducing kernel Hilbert spaces for penalized regression: a tutorial. The American Statistician. 2012; 66:50–60.

47. Osher S, Burger M, Goldfarb D, Xu J, Yin W. An iterative regularization method for total variation-based image restoration. Multiscale Modeling and Simulation. 2005; 4:460–489.

48. Park SC, Park MK, Kang MG. Super-Resolution image reconstruction: a technical overview. IEEE signal processing magazine. 2003; 20:21–36.

49. Quang MH, Kang SH, Le TM. Image and video colorization using vector-valued reproducing kernel Hilbert spaces. Journal of Mathematical Imaging and Vision. 2010; 37:49–65.

50. Schökopf, B., Smola, A. Learning with kernels: support vector machines, regularization, optimization, and beyond. MIT Press; Cambridge: 2002.

51. Seaman R, Hutchinson M. Compamtive real data tests of some objective analysis methods by withholding. Australian Meteorological Magazine. 1985; 33:37–46.

52. Shah AJ, Gupta SB. Image super resolution - a survey. International Conference on Emerging Technology Trends in Electronics, Communication and Networking. 2012

53. Shan Q, Li Z, Jia J, Tang C. Fast Image/Video Upsampling. ACM Transactions on Graphics (TOG). 2008; 27

54. Shawe-Taylor, J., Cristianini, N. Kernel methods for pattern analysis. Cambridge University Press; Cambridge: 2004.

55. Sun J, Sun J, Xu Z, Shum H-Y. Image super-resolution using gradient profile prior. CVPR. 2008:1–8.

56. Sun J, Zheng NN, Tao H, Shum H. Image hallucination with primal sketch priors. IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2003; 2:729–736.

57. Tai S-C, Kuo T-M, Iao C-H, Liao T-W. A fast algorithm for single-image super resolution in both wavelet and spatial domain. International Symposium on Computer, Consumer and Control. 2012:702–705.

58. Tai Y-W, Liu S, Brown M, Lin S. Super resolution using edge prior and single image detail synthesis. CVPR. 2010:2400–2407.

59. Takeda H, Farsiu S, Milanfar P. Kernel regression for image processing and reconstruction. IEEE Transactions on Image Processing. 2007; 16:349–366. [PubMed: 17269630]

60. Tappen MF, Russell BC, Freeman WT. Exploiting the sparse derivative prior for super-resolution and image demosaicing. IEEE Workshop on Statistical and Computational Theories of Vision. 2003

61. Van Ouwerkerk JD. Image super-resolution survey. Image and Vision Computing. 2006; 24:1039–1052.

62. Viola F, Fitzgibbon AW, Cipolla R. A unifying resolution-independent formulation for early vision. CVPR. 2012:494–501.

63. Wahba G. Spline models for observational data. SIAM CBMS-NSF Regional Conference Series in Applied Mathematics. 1990; 59

64. Wahba G, Wendelberger J. Some new mathematical methods for variational objective analysis using splines and cross-validation. Monthly Weather Review. 1980; 108:1122–1145.

65. Wang L, Wu H, Pan C. Fast image upsampling via the displacement field. IEEE Trans Image Processing. 2014; 23:5123–5135.

66. Wang L, Xiang S, Meng G, Wu H, Pan C. Edge-Directed single-image super-resolution via adaptive gradient magnitude self-interpolation. IEEE Trans Circuits and Systems for Video Technology. 2013; 23:1289–1299.

67. Wang Y, Yang J, Yin W, Zhang Y. A new alternating minimization algorithm for total variation image reconstruction. SIAM Journal on Imaging Sciences. 2008; 1:248–272.

68. Wang Z, Bovik AC, Sheikh HR, Simoncelli EP. Image quality assessment: From error visibility to structural similarity. IEEE Trans Image Processing. 2004; 13:600–612.

69. Xie Q, Chen H, Cao H. Improved example-based single-image superresolution. International Congress on Image and Signal Processing (CISP). 2010; 3:1204–1207.

70. Yang J, Wang Z, Zhe L, Huang T. Coupled dictionary training for image super-resolution. IEEE transactions on image processing. 2011; 21:3467–3478.

71. Yang J, Wright J, Huang T, Ma Y. Image super-resolution via sparse representation. IEEE transactions on image processing. 2010; 19:2861–1873. [PubMed: 20483687]

72. Yang J, Wright J, Ma Y, Huang T. Image super-resolution as sparse representation of raw image patches. IEEE Conference on Computer Vision and Pattern Recongnition (CVPR). 2008:1–8.

73. Zeyde R, Elad M, Protter M. On single image scale-up using sparse-representations. Curves and Surfaces, Lecture Notes in Computer Science. 2012; 6920:711–730.

74. Zhang L, Wu X. An Edge-Guided Image Interpolation Algorithm via Directional Filtering and Data Fusion. IEEE Trans Image Processing. 2006; 15:2226–2238.

75. Zhao Y, Yang J, Zhang Q, Lin S, Cheng Y, Pan Q. Hyperspectral imagery superresolution by sparse representation and spectral regularization. EURASIP Journal on Advances in Signal Processing. 2011

76. Zheng H, Bouzerdoum A, Phung SL. Wavelet based nonlocal-means superresolution for video sequences. IEEE International Conference on Image Processing (ICIP). 2010:2817–2820.
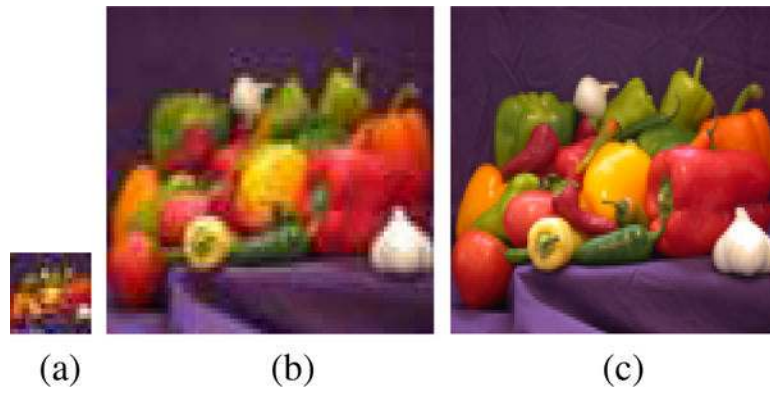
Fig. 1.
(a) A low-resolution image; (b) The super-resolution image using bicubic interpolation method (note the blur effect); (c) The super-resolution image by nearest-neighbor interpolation method (note the jaggy effect on the edges); The upscaling factor is 4.
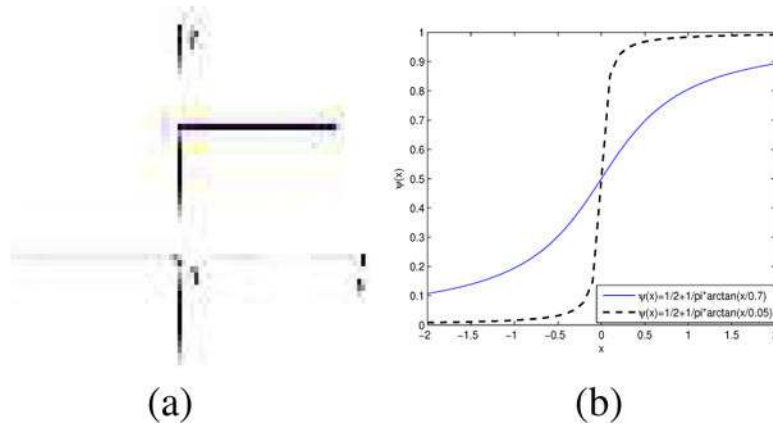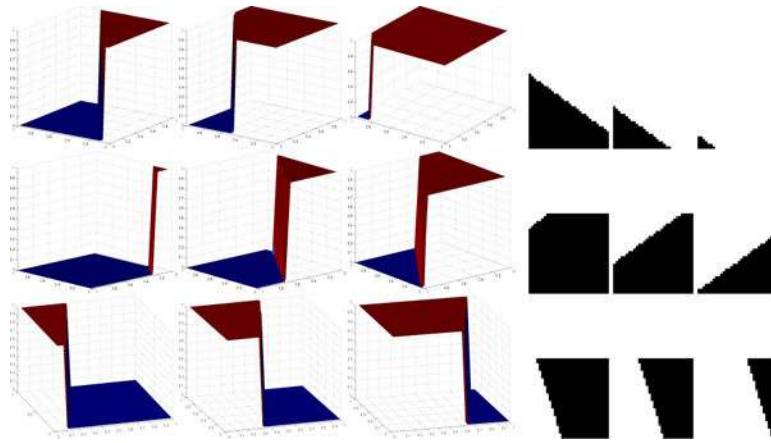
Fig. 2.
(a) 1D Heaviside function; (b) Two approximated Heaviside functions with $\xi = 0.7$ (blue solid line) and $\xi = 0.05$ (black dash line), respectively; The smaller ξ the sharper edge (color images are better visualized in the pdf file).

Fig. 3.
Left panel: 3D surface images of ψ under $\xi = 10^{-4}$ and nine parameter pairs (θ, c); right panel: the corresponding 2D images. From left to right and then from top to bottom:

$$\left(\frac{4\pi}{5}, \frac{51}{1024}\right), \left(\frac{4\pi}{5}, \frac{25}{64}\right), \left(\frac{4\pi}{5}, \frac{175}{256}\right), \left(\frac{6\pi}{5}, \frac{135}{1024}\right), \left(\frac{6\pi}{5}, \frac{1}{2}\right), \left(\frac{6\pi}{5}, \frac{25}{32}\right), \left(\frac{8\pi}{5}, \frac{5}{64}\right),$$

$\left(\frac{8\pi}{5}, \frac{75}{256}\right), \left(\frac{8\pi}{5}, \frac{75}{128}\right)$, (for better visualization, some 3D surface images in the left panel are rotated so that we can observe the edge jumps clearly).
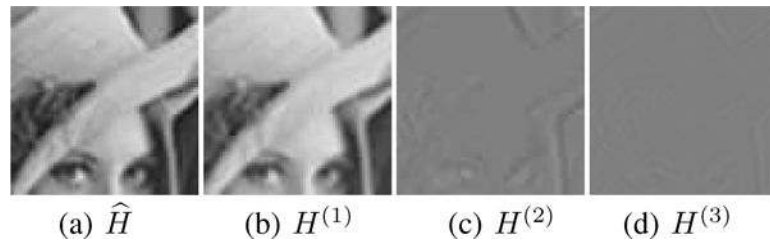
(a) $\widehat{H}$ (b) $H^{(1)}$ (c) $H^{(2)}$ (d) $H^{(3)}$

Fig. 4.
Super-resolution image "lena" by Algorithm 2; (a) is the sum image of $H^{(i)}$, $i = 1, 2, 3$; (b) is the computed image for first iteration. For better vision, we add 0.5 to the intensities of $H^{(2)}$ and $H^{(3)}$ to obtain (c) and (d), respectively. From last two images, we know that $H^{(2)}$ and $H^{(3)}$ pick up some image details.
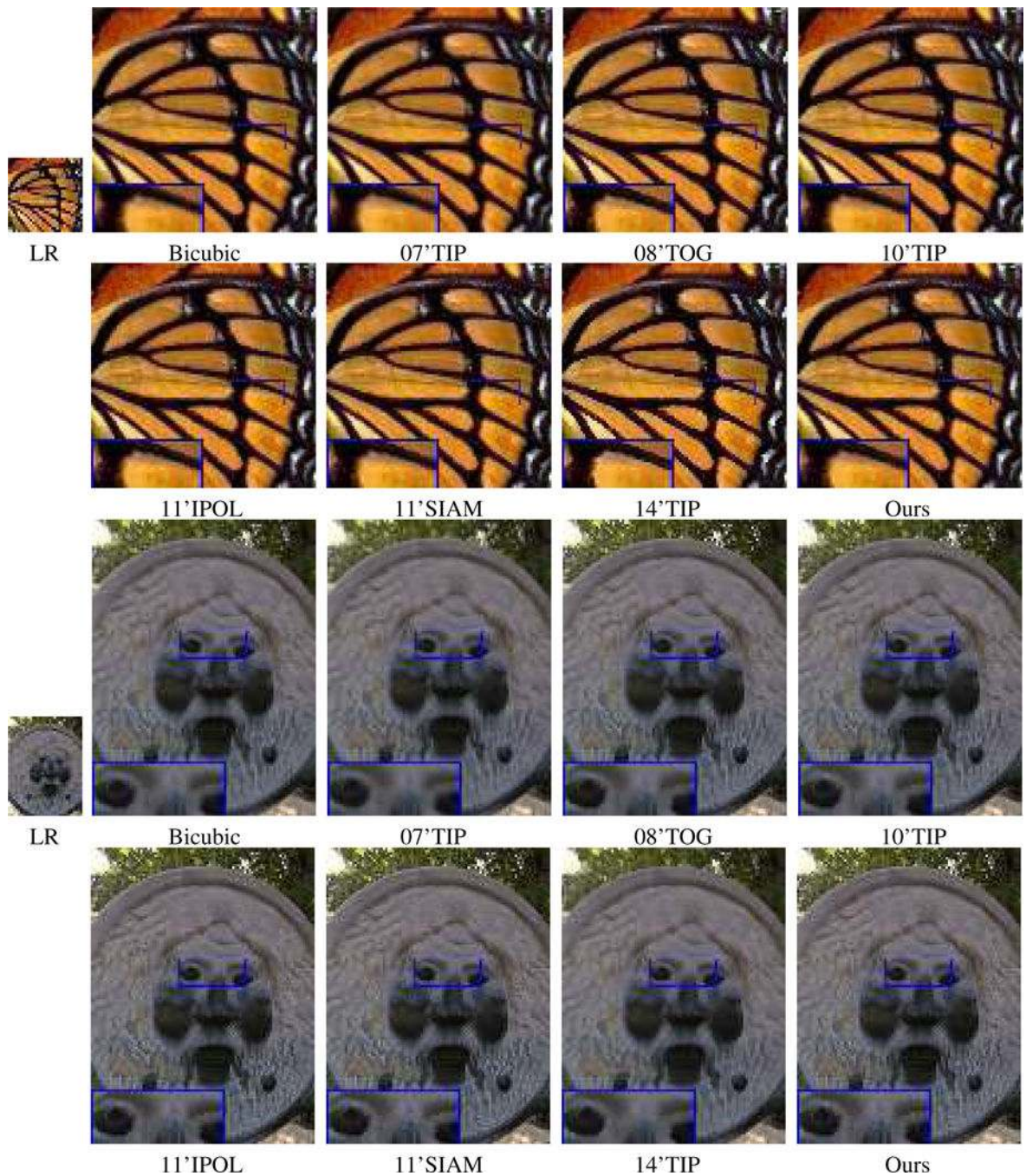
Fig. 5.
Compare the proposed algorithm with some state-of-the-art approaches: Bicubic interpolation, 07'TIP [59], 08'TOG [53], 10'TIP [71], 11'IPOL [27], 11'SIAM [26] and 14'TIP [65]. The upscaling factor is 3. No ground-truth high-resolution images are available for quantitative comparison. Color images are better visualized in the pdf file.
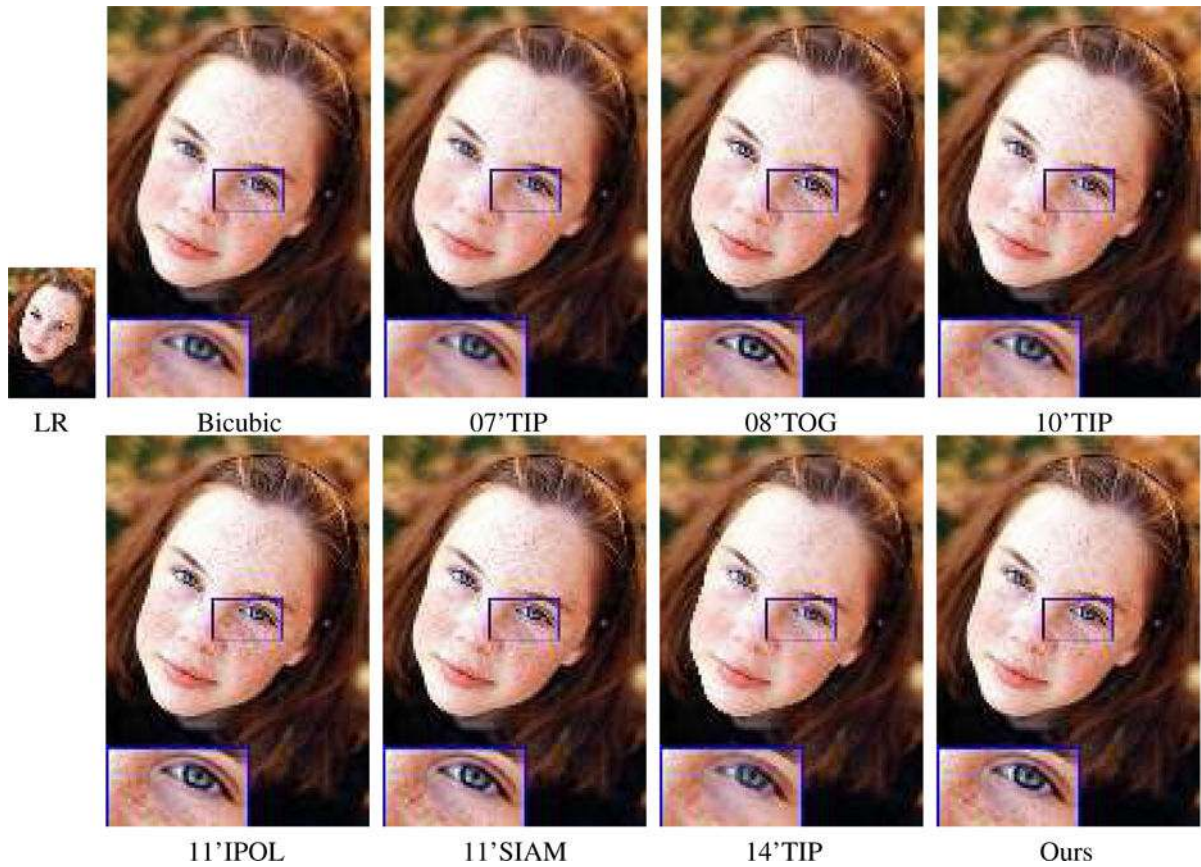
Fig. 6.
Compare the proposed algorithm with some state-of-the-art approaches: Bicubic interpolation, 07'TIP [59], 08'TOG [53], 10'TIP [71], 11'IPOL [27], 11'SIAM [26] and 14'TIP [65]. The upscaling factor is 3. No ground-truth high-resolution images are available for quantitative comparison.
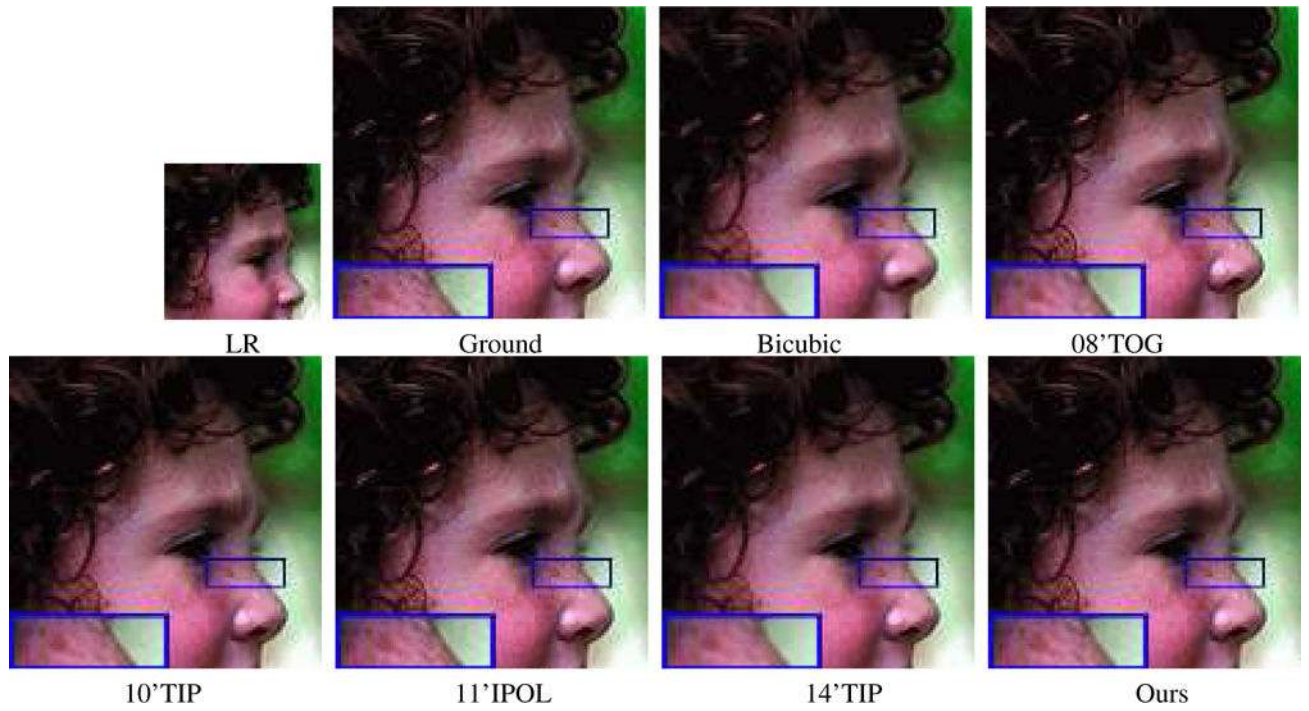
Fig. 7.
Qualitative comparison for the image "face" among the proposed method and Bicubic,
"08'TOG" [53], "10'TIP" [71], "11'IPOL" [27] and "14'TIP" [65], with the upscaling factor
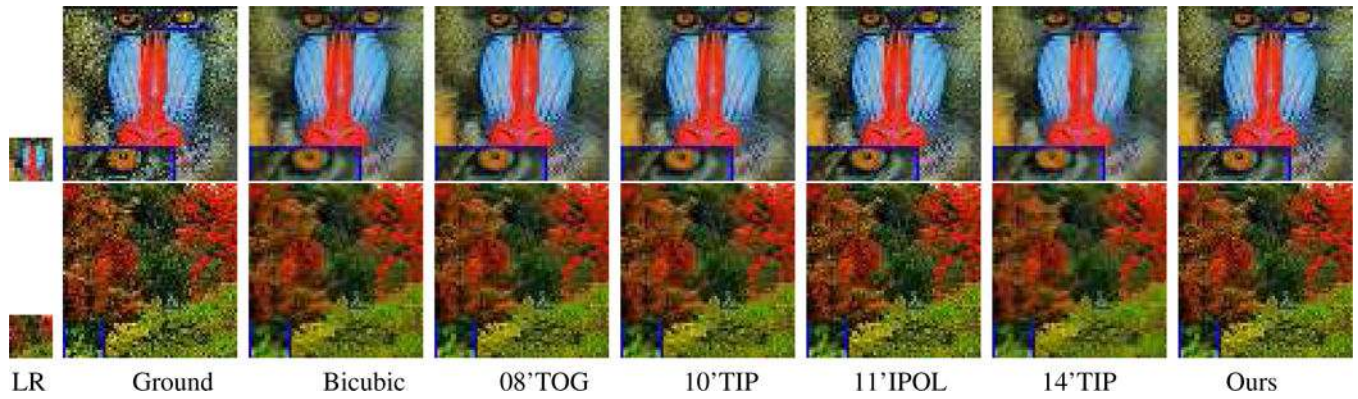of 2.

| LR | Ground | Bicubic | 08'TOG | 10'TIP | 11'IPOL | 14'TIP | Ours |

Fig. 8.
Results of "baboon" (upscaling factor 4) and "forest" (upscaling factor 4); Compared methods: Bicubic interpolation, "08'TOG" [53], "10'TIP" [71], "11'IPOL" [27] and "14'TIP" [65] and ours. In particular, readers are recommended to zoom in all figures for better visualization.
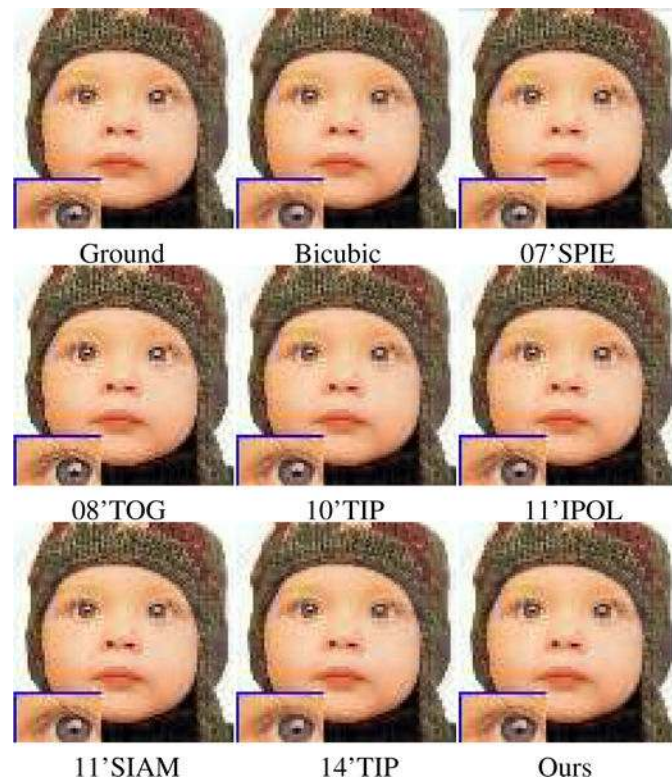
Fig. 9.
Results of "baby" with the upscaling factor of 2; First row: Ground-truth image, Bicubic interpolation (RMSE = 3.58; PSNR = 37.06; SSIM = 0.993), "07'SPIE" [45] (3.73; 36.70; 0.996); Second row: "08'TOG" [53] (4.32; 35.43; 0.982), "10'TIP" [71] (3.40; 37.51; 0.995), "11'IPOL" [27] (3.37; 37.58; 0.997); Third row: "11'SIAM" [26] (3.24; 37.93; 0.997), "14'TIP" [65] (4.19; 36.82; 0.985) and the proposed method (3.17; 38.19; 0.997).
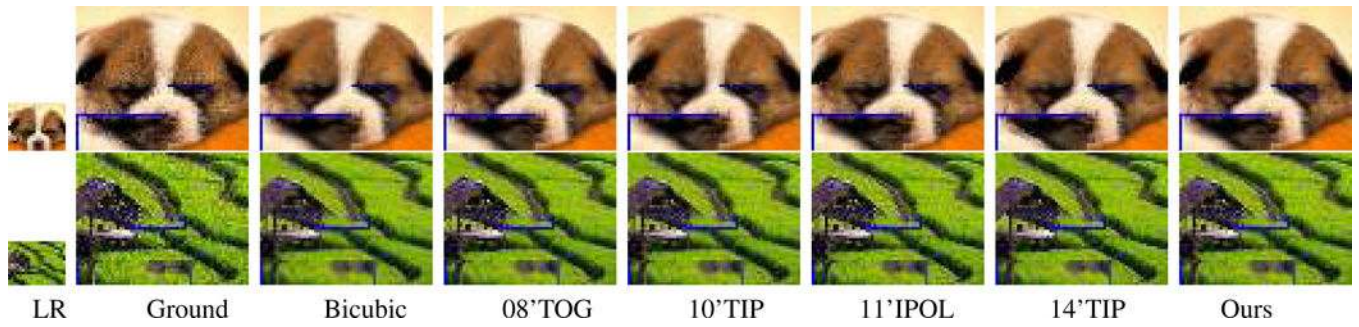
LR | Ground | Bicubic | 08'TOG | 10'TIP | 11'IPOL | 14'TIP | Ours

Fig. 10.
Results of "dog" (upscaling factor 3) and "field" (upscaling factor 3); Compared methods: Bicubic interpolation, "08'TOG" [53], "10'TIP" [71], "11'IPOL" [27] and "14'TIP" [65] and the proposed method.
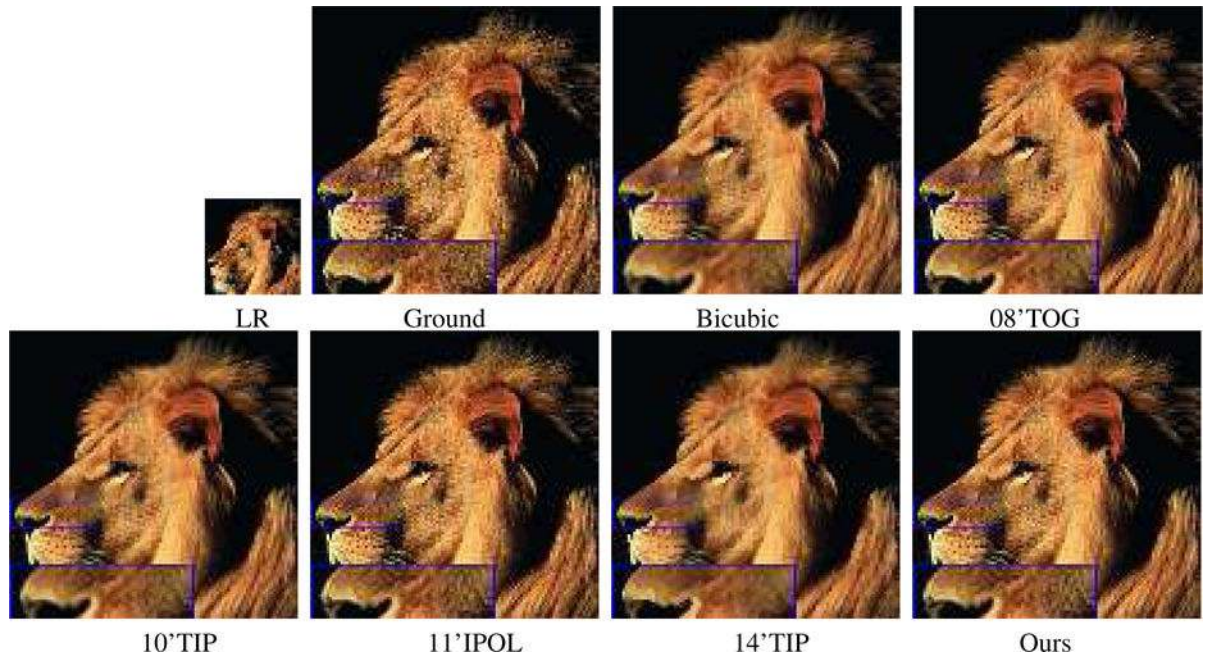
Fig. 11.
Results of "lion" with the upscaling factor of 3; Compared methods: Bicubic interpolation, "08'TOG" [53], "10'TIP" [71], "11'IPOL" [27] and "14'TIP" [65] and the proposed method.
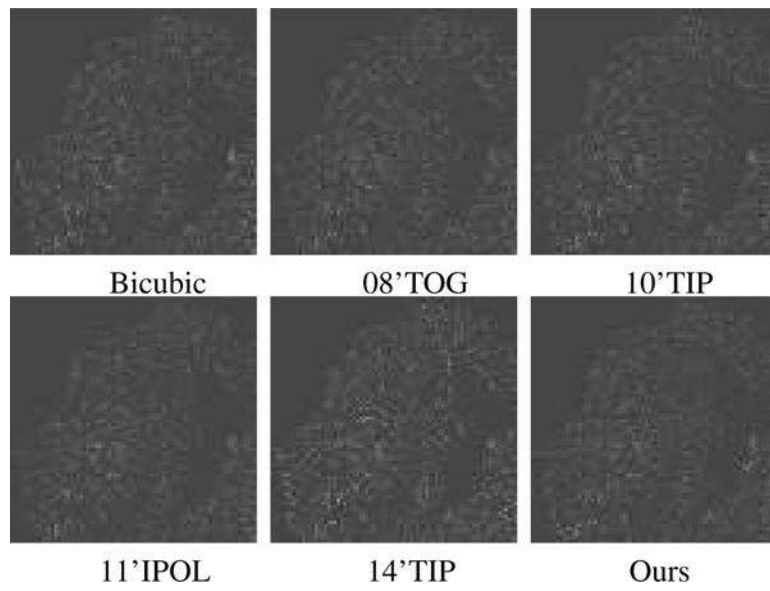
Fig. 12.
Compare error maps of the proposed method and five other methods. The test image is "lion". The error maps are brightened for better visualization.
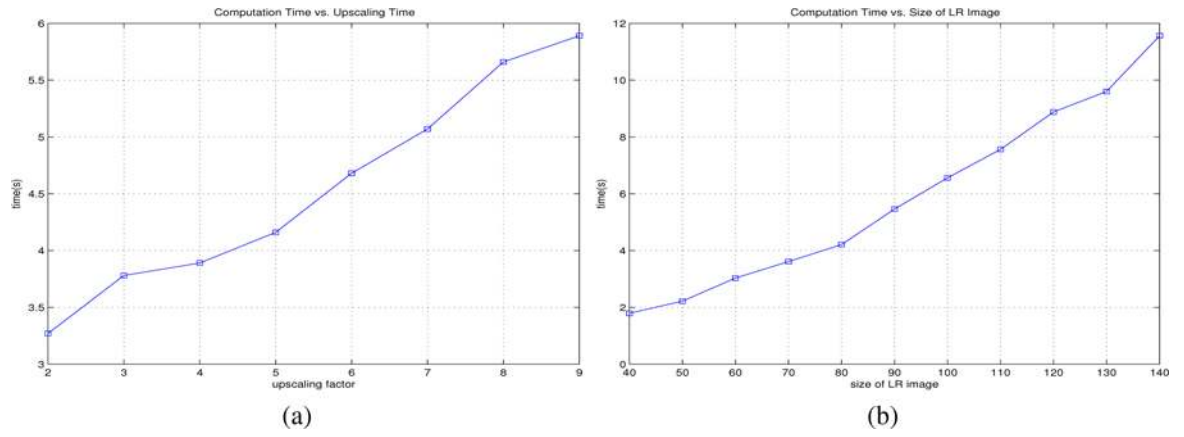
Fig. 13.
(a) Computation time vs. upscaling factor for low-resolution image with size 80×80; (b) Computation time vs. size of low-resolution image, the size of low-resolution image is from 40×40 to 140×140 and the upscaling factor is always 5.

Fig. 14.
(a) Result of model (20) with the iterative strategy (i.e., $\tau = 3$ in Algorithm 2, RMSE = 10.25); (b) Result of model (20) without the iterative strategy (i.e., $\tau = 1$ in Algorithm 2, RMSE = 10.36); (c) Error map between (a) and (b). Upscaling factor: 3.

Algorithm 1

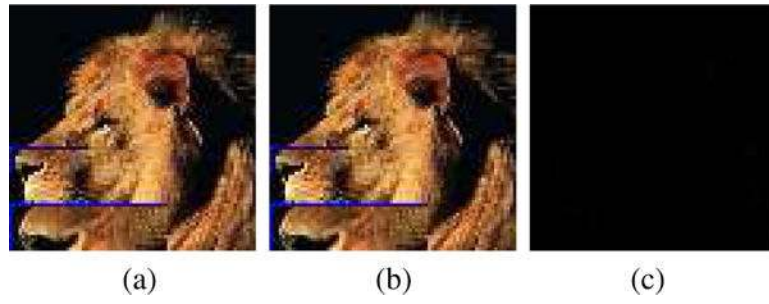**Input:** Given L, T^l, K^l, Ψ^l, λ, a, ρ, $\gamma \in \left(0, \left(\sqrt{5}+1\right)/2\right)$

Output: c, d, β

$j \leftarrow 0$, $(c^{(j)}, d^{(j)}, \beta^{(j)}) \leftarrow 0$, $u^{(j)} \leftarrow 0$, $b^{(j)} \leftarrow 0$

while not converged do

    1. $j \leftarrow j + 1$

    2. $(c^{(j)}, d^{(j)}, \beta^{(j)}) \leftarrow$ solve subproblem (23) for $u = u^{(j-1)}$, $b = b^{(j-1)}$

    3. $u^{(j)} \leftarrow$ solve subproblem (24) for $\beta = \beta^{(j)}$, $b = b^{(j-1)}$

    4. $b^{(j)} \leftarrow b^{(j-1)} + \gamma(u^{(j)} - \beta^{(j)})$

End while.

Algorithm 2

(Single image super-resolution via RKHS (SR-RKHS))

Input: one low-resolution image $L \in \mathbb{R}^{Q \times S}$, $\lambda > 0$, $a > 0$, τ: maximum number of iteration

Output: high-resolution image $\hat{H} \in \mathbb{R}^{U \times V}$

Step 1. Set coarse grids t^l and fine grids t^h

Step 2. Construct matrices T^l, K^l, Ψ^l (refer to Section II-C and Section III-A) for $i, j = 1, 2, \cdots, n$, $n = Q.S$, $\nu = 1, 2, \cdots, M$. Similarly for T^h, K^h, Ψ^h except $i = 1, 2, \cdots, N$; $j = 1, 2, \cdots, n$, where $N = U \cdot V$

Step 3. Initialization: $L^{(1)} = L$.

for k = 1: τ

a. Compute the coefficients:
$$\left(c^{(k)}, d^{(k)}, \beta^{(k)}\right) = \arg\min \frac{1}{n} \left\| L^{(k)} - DB \left(T^h d + K^h c + \Psi^h \beta\right) \right\|^2 + \lambda c' K^l c + \alpha \|\beta\|_1 .$$

b. Update the high-resolution image: $H^{(k)} = T^h d^{(k)} + K^h c^{(k)} + \Psi^h \beta^{(k)}$.

c. Down-sampling $H^{(k)}$ to coarse grid: $\tilde{L} = DBH^{(k)}$.

d. Compute residual: $L^{(k+1)} = L^{(k)} - \tilde{L}$.

end

Step 4. Compute the final high-resolution image: $\hat{H} = \sum_{i=1}^{\tau} H^{(i)}$.

TABLE I

Quantitative Comparisons for Different Methods in Terms of RMSE, PSNR and SSIM (Bold: the Best one; Underline: the Second Best). Compared Methods: Bicubic Interpolation, "08'TOG" [53], "10'TIP" [71], "11'IPOL" [27] AND "14'TIP" [65] and the Proposed Method.

Image(factor)	Index	Bicubic	08'TOG	10'TIP	11'IPOL	14'TIP	Proposed
face(X2)	RMSE	4.61	5.02	4.41	4.38	5.55	**4.26**
	PSNR	34.86	34.11	35.24	35.30	33.25	**35.53**
	SSIM	0.862	0.843	0.873	0.879	0.828	**0.883**
baboon(X4)	RMSE	19.47	19.29	19.32	19.22	19.51	**19.13**
	PSNR	22.25	22.31	22.30	22.41	21.88	**22.43**
	SSIM	0.704	0.714	0.718	**0.760**	0.711	0.756
forest(X4)	RMSE	18.63	18.30	18.47	18.44	18.84	**18.09**
	PSNR	22.78	22.93	22.85	22.86	22.79	**23.03**
	SSIM	0.685	0.701	0.701	0.738	0.646	**0.748**
baby(X2)	RMSE	3.58	4.32	3.40	3.37	4.19	**3.17**
	PSNR	37.06	35.43	37.51	37.58	36.82	**38.19**
	SSIM	0.993	0.982	0.995	**0.997**	0.985	**0.997**
dog(X3)	RMSE	9.15	9.24	9.09	9.11	9.99	**9.02**
	PSNR	28.90	28.82	28.96	28.94	28.14	**29.04**
	SSIM	0.914	0.904	0.920	0.927	0.893	**0.928**
field(X3)	RMSE	12.49	12.32	12.27	12.22	13.74	**12.06**
	PSNR	26.20	26.32	26.36	26.39	25.37	**26.51**
	SSIM	0.600	0.595	0.615	0.638	0.559	**0.641**
lion(X3)	RMSE	11.03	10.66	10.67	10.30	11.58	**10.25**
	PSNR	27.28	27.58	27.57	27.87	27.13	**27.89**
	SSIM	0.883	0.894	0.897	**0.916**	0.874	**0.916**

TABLE II

Time Comparison for Different Methods (Bold: the Best One; Underline: the Second Best). Compared Methods: Bicubic Interpolation, "08'TOG" [53], "10'TIP" [71], "11'IPOL" [27], "14'TIP" [65] and the Proposed Method. Note that Bicubic is Optimized in MATLAB, "08'TOG" is Optimized By an Executable Software, "11'IPOL" and "14'TIP" are Speeded Up Via C Language and Cmex, Respectively. Only "10'TIP" and the Proposed Method are Based on MATLAB Codes that are not Optimized. (Unit: Second)

Image(factor)	size of LR	Bicubic	08'TOG	10'TIP	11'IPOL	14'TIP	Proposed
face(X2)	140 × 140	0.01	3.32	40.70	0.09	0.93	9.54
baboon(X4)	120 × 120	0.02	9.11	145.76	0.26	1.10	8.20
tree(X4)	110 × 110	0.02	5.42	115.16	0.23	1.03	7.74
baby(X2)	256 × 256	0.05	13.39	144.67	0.37	1.00	43.59
dog(X3)	130 × 140	0.03	11.00	100.32	0.19	0.81	12.61
field(X3)	100 × 133	0.02	9.00	81.54	0.14	0.67	9.65
lion(X2)	168 × 168	0.03	13.98	126.49	0.25	0.86	13.75