

Edge-Directed Single-Image Super-Resolution via Adaptive Gradient Magnitude Self-Interpolation

Lingfeng Wang, Shiming Xiang, Gaofeng Meng, Huaiyu Wu, and Chunhong Pan

Abstract—Super-resolution from a single image plays an important role in many computer vision systems. However, it is still a challenging task, especially in preserving local edge structures. To construct high-resolution images while preserving the sharp edges, an effective edge-directed super-resolution method is presented in this paper. An adaptive self-interpolation algorithm is first proposed to estimate a sharp high-resolution gradient field directly from the input low-resolution image. The obtained high-resolution gradient is then regarded as a gradient constraint or an edge-preserving constraint to reconstruct the high-resolution image. Extensive results have shown both qualitatively and quantitatively that the proposed method can produce convincing super-resolution images containing complex and sharp features, as compared with the other state-of-the-art super-resolution algorithms.

Index Terms—Edge-directed, gradient magnitude transformation, super-resolution.

I. INTRODUCTION

SUPER-RESOLUTION is a fundamental task for various computer vision applications, such as object detection and recognition, video compression, and communication. Generally, the goal of image super-resolution methods is to recover a high-resolution (HR) image from one or a sequence of low-resolution (LR) images [1]–[20]. In many real-world applications, only one LR image is available. Hence, super-resolution from a single-input LR image is practically in great need. Unfortunately, the problem of single-image super-resolution is under-constrained, which makes classical methods, such as interpolation or reconstruction-based methods, often bring undesired artifacts in the HR image, especially along the salient edges. To overcome this limitation or to preserve local edge structures in the HR image, it is effective to introduce an additional edge constraint, which is the main issue that we will address in this paper.

The problem of single-image super-resolution has received much attention from both computer graphic and computer

vision communities. Generally, approaches addressing this problem can be grouped into four categories [21], [12], i.e., the interpolation-based, learning-based, reconstruction-based, and edge-directed methods.

The interpolation-based methods [8], [13], [18], [19], [22] are widely used for producing zoom-in images because of its simplicity. Unfortunately, these solutions tend to produce visual artifacts, such as ringing, aliasing, blocking, and blurring, especially on salient edges. A survey of these methods and their shortcomings is given in [19] in detail.

The learning-based or example-based super-resolution methods are first introduced in [17], [20], and [23], and extended later by others, such as [6], [7], [15], and [24]. In these methods, the correspondences between LR and HR image patches are first learned from a database of LR and HR image pairs, and then applied to a new LR image to recover its HR version. Improved performances compared with traditional super-resolution methods have been reported in these papers. However, the learning-based methods rely largely on the quality of the prior used, that is, the similarity between the training set and the test set. Furthermore, the computational cost of these methods is expensive due to a large training set being used.

The reconstruction-based methods [25]–[27] emphasize the reconstruction constraint, which requires that the down-sampling version of the target HR image should be close to the LR image. However, the main problem is that the reconstructed edges are sometimes too sharp and look unnatural. Moreover, these methods often introduce undesired artifacts, such as ringing, in the HR image, especially along salient edges.

The sophisticated methods based on the edge models (edge-directed) have been proposed in [5], [10], [11], [14], [21], [28], and [29]. These methods estimate the target HR image by enforcing some edge knowledge, such as the smooth edge [10], [11], [28] or the gradient profile prior (GPP) [5], [21], [29]. The enforced edge knowledge is able to produce sharp edges with minimal jaggy or ringing artifacts. Unfortunately, these two edge models have their own drawbacks. First, the smooth edge knowledge is a smoothness constraint. Hence, some small scale information cannot be well recovered in the HR image. That is, some image details, especially on the weak edges, may be hard to recover in the HR image. Second, in [29], the super-resolution algorithm needs to locate edge pixels. Therefore, their results partially rely on the edge pixel location accuracies. The gradient profile prior is learned from an image dataset, so the image super-resolution result of [29]

Manuscript received March 15, 2012; revised July 16, 2012 and September 19, 2012; accepted October 8, 2012. Date of publication January 17, 2013; date of current version July 31, 2013. This work was supported in part by the National Basic Research Program of China, under Grant 2012CB316304, and by the National Natural Science Foundation of China under Grants 61175025, 61005036, and 61272331. This paper was recommended by Associate Editor W. Zeng.

The authors are with the Department of National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China (e-mail: lfwang@nlpr.ia.ac.cn; smxiang@nlpr.ia.ac.cn; gfmeng@nlpr.ia.ac.cn; hywu@nlpr.ia.ac.cn; chpan@nlpr.ia.ac.cn).

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

Digital Object Identifier 10.1109/TCSVT.2013.2240915

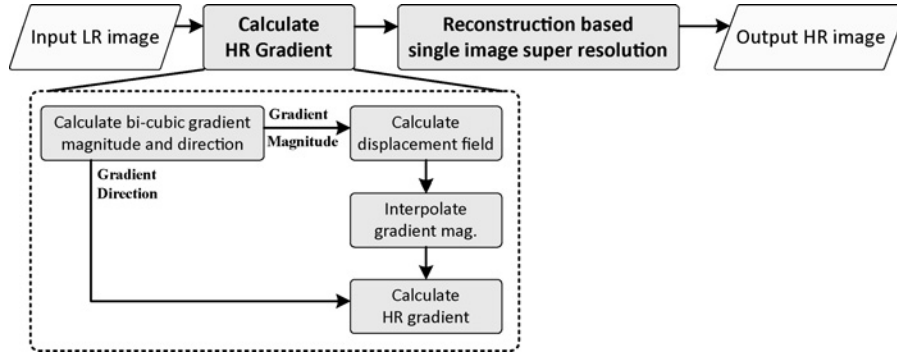


Fig. 1. Overview of our super-resolution algorithm. First, a HR gradient is estimated by adaptive gradient magnitude self-interpolation. Then, the HR image is recovered by the reconstruction based super-resolution framework (see Sections II and VI). In the dashed box, we present the details about HR gradient estimation. The gradient magnitude is first used to compute a displacement field (refer to Section V), which is, thereafter, adopted to sharpen the gradient magnitude by an interpolating method (refer to Section IV).

relies on the utilization of training dataset to some extent. Third, the two types of methods are under the edge-directed reconstruction framework. In this framework, an iteration method is utilized to solve the objective functional. Hence, a good initialization is necessary on both reducing the iteration times and recovering the HR image. However, these methods often adopt a bicubic up-sampled image, which is blurred in the edges as the initialization.

Motivated by the previous work [10], [25], [29], we propose an edge-directed super-resolution approach based on a novel adaptive gradient magnitude self-interpolation. The mainflow of our algorithm is shown in Fig. 1. We estimate a sharp HR gradient field by an adaptive self-interpolation algorithm. In our self-interpolation algorithm, the displacement field is calculated from the gradient magnitude of the bicubic up-sampled image. It is worth noting that this displacement field is also used to interpolate the bicubic up-sampled image as an initialization. The estimated HR gradient is used as a gradient constraint or a edge-preserving constraint to recover the HR image. The main advantages or details of our algorithm are summarized as follows.

- 1) The proposed super-resolution method introduces an additional edge constraint to reduce the undesired artifacts brought by the traditional algorithm. Hence, compared with the classical interpolation and reconstruction-based methods, our algorithm can effectively get rid of some visual artifacts, such as ringing, aliasing, and blurring.
- 2) Different from the smoothness edge methods, such as the soft edge smoothness proposed in [11], our self-interpolation sharpening method does not belong to the smoothness constraint. Hence, both small and large scales can be better recovered in our HR image, as compared with the smoothness edge methods.
- 3) Compared with the gradient magnitude sharpening method proposed in [29], our method has two advantages.

The gradient sharpening algorithm used in [29] depends on the gradient profile prior learnt from an image dataset. However, our adaptive self-interpolation algorithm relies on the displacement field, which is estimated from the input LR image directly. That is, our method

does not rely on the using of training datasets. Furthermore, in [29], it needs to locate the edge-pixels before sharpening the gradient, while our method interpolates the whole image domain directly to obtain the sharp gradient. Hence, our method is more general.

The computational cost of our method is lower. GPP edge-directed [29] and our method both have two time-consuming procedures, i.e., the gradient sharpening and the reconstruction. In the gradient sharpening, the experimental results show that our method is three times faster than [29]. In the reconstruction, the displacement field estimated by our method can also be used to obtain a sharp image as initialization. Comparing the bicubic up-sampled image used in [29], our sharp initialization image is more close to the desired image. Hence, less iterations are needed in our method. In practice, 30 iterations are enough for our method, while 100 iterations are used in [29]. A smaller iteration makes the computational cost lower.

The remainder of this paper is organized as follows. A brief introduction of an edge-directed single-image super-resolution framework is presented in Section II. The main motivation on HR gradient field estimation is presented in Section III. The gradient sharpening method is presented in Section IV. The adaptive gradient magnitude self-interpolation algorithm, especially the displacement field estimation, is described in Section V. The HR image is reconstructed by optimization of an objective energy functional, which is presented in Section VI. Section VII gives the experimental results. Conclusions and future work are given in Section VIII.

II. EDGE-DIRECTED SINGLE-IMAGE SUPER-RESOLUTION

Within the reconstruction framework, the goal of edge-directed image super-resolution is to estimate the HR image I_h based on two inputs, i.e., the LR image I_l and the HR gradient field $\widehat{\nabla I_h}$. The estimation of I_h^* can be formulated as the minimization of the following energy functional:

$$\begin{aligned}
 I_h^* &= \arg \min_{I_h} E(I_h | I_l, \widehat{\nabla I_h}) \\
 &= \arg \min_{I_h} E_d(I_h | I_l) + \alpha E_g(\nabla I_h | \widehat{\nabla I_h})
 \end{aligned} \quad (1)$$

where $E_d(\cdot)$ and $E_g(\cdot)$ are the down-sampling energy and the gradient energy, respectively, and α is the weighting constant to balance these two energies. The gradient field ∇I is denoted as $\nabla I = (\partial_x I, \partial_y I) = G \cdot \theta$, where $G = \sqrt{(\partial_x I)^2 + (\partial_y I)^2}$ is the gradient magnitude and $\theta = \arctan(\partial_y I / \partial_x I)$ is the gradient direction. To obtain $\partial_x I$ and $\partial_y I$, the image I is convolved by the discrete gradient operators $[-1/2 \ 0 \ 1/2]$ and $[-1/2 \ 0 \ 1/2]^T$, respectively. The down-sampling energy (or constraint) is represented by the difference between the LR image I_l and the down-sampled version of the HR image I_h

$$E_d(I_h | I_l) = \|[I_h \otimes g]_{\downarrow(\beta)} - I_l\|^2$$

where \otimes is the convolution operation, $[\cdot]_{\downarrow(\beta)}$ is the down-sampling operation with factor β , and g is the blurry kernel. The gradient constraint requires the gradient of the HR image ∇I_h should be close to the input gradient $\widehat{\nabla I_h}$

$$E_d(\nabla I_h | \widehat{\nabla I_h}) = \|\nabla I_h - \widehat{\nabla I_h}\|^2.$$

Combining with (1), the edge-directed image super-resolution estimates the HR image I_h^* by minimizing the following objective energy functional:

$$I_h^* = \arg \min_{I_h} \|[I_h \otimes g]_{\downarrow(\beta)} - I_l\|^2 + \alpha \|\nabla I_h - \widehat{\nabla I_h}\|^2. \quad (2)$$

From (2), we see that the estimation of I_h mainly relies on two aspects, namely, the blurry kernel g and the HR gradient field $\widehat{\nabla I_h}$. In this paper, we focus on the second aspect, i.e., the estimation of the HR gradient field $\widehat{\nabla I_h}$. The blurry kernel is assumed to be a 2-D Gaussian kernel, and its standard variance is related to the up-sampling factors, i.e., it is set to 1.2, 1.5, and 1.8 for the up-sampling factors of 2, 3, and 4.

III. MOTIVATION OF GRADIENT SHARPENING

The estimation accuracy of the HR gradient field has both theoretical and practical importance on the edge-directed single-image super-resolution problem. As shown in (2), without a gradient constraint, that is, only using the first term, the super-resolution problem is severely under-constrained. Adding a gradient constraint can make this problem easy to solve (determined or even overdetermined). It is worth noting that the reconstruction-based method, such as back-projection [25], only considers the down-sampling energy, ignoring the gradient constraint. Theoretically, the back-projection algorithm is convergent. That is, the output HR image obtained by back-projection converges to a desired image that satisfies that the reconstruction error (or the down-sampling energy) is close to zero. However, this algorithm often brings undesired artifacts in the HR image, especially along salient edges. Hence, it is important to introduce additional edge constraints to the traditional reconstruction-based method.

IV. GRADIENT SHARPENING

In the actual super-resolution problem, we only have the input LR image, but do not have the true HR gradient field. Hence, a more wise and effective method is to estimate the

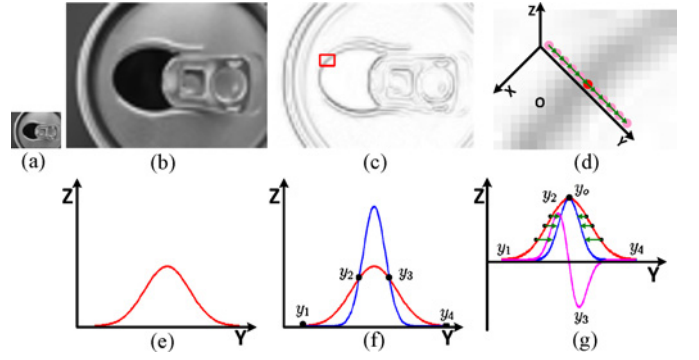


Fig. 2. Example of sharpening gradient magnitude. (a) Input LR image. (b) Bicubic up-sampled image (3X magnification). (c) Gradient maps (normalized and inverted magnitudes) of (b). (d) Closeups of the red rectangular region. Note that the X - O - Y plane is the image plane, and the z -axis represents the gradient magnitude. The green arrows illustrate a 1-D path passing through the edge-pixel (see the red point). (e) Gradient profile is the curve of gradient magnitude along the green arrows of (d). (f) Gradient magnitude sharpened by GPP [21]. The red curves in (f) and (g) are the original gradient magnitudes, while the blue curves are the sharpened gradient magnitudes. (g) Gradient magnitude sharpened by our method. The green arrows in (g) are the displacement field, while the pink curve is the gradient of the original gradient profile (red curve).

$\widehat{\nabla I_h}$ from the LR image, such as the gradient of the bicubic up-sampled LR image, i.e., ∇I_h^u , where

$$I_h^u = [I_l]_{\uparrow(\beta)}$$

in which $[\cdot]_{\uparrow(\beta)}$ is the (bicubic) up-sampling operation and β is the down-sampling factor. In practical, the bicubic up-sampled image I_h^u will be blurred in the image edge, which makes the corresponding gradient magnitude blurry [refer to Fig. 2(b), (c)]. Hence, it is infeasible to directly use ∇I_h^u . Fortunately, we can adopt its transformation version to represent $\widehat{\nabla I_h}$. That is, $\widehat{\nabla I_h}$ is calculated as follows:

$$\widehat{\nabla I_h} = \text{Tran}(\nabla I_h^u) \quad (3)$$

where $\text{Tran}(\cdot)$ is a transformation function. We denote the HR gradient field $\widehat{\nabla I_h}$ as $\widehat{\nabla I_h} = \widehat{G}_h \cdot \widehat{\theta}_h$. Generally, to obtain the sharp gradient $\widehat{\nabla I_h}$, we only need to sharpen its magnitude \widehat{G}_h , while the direction $\widehat{\theta}_h$ can be estimated by the bicubic up-sampled version, i.e., $\widehat{\theta}_h = \theta_h^u$, where θ_h^u is the gradient direction of ∇I_h^u . Hence, (3) is simplified as

$$\widehat{G}_h = \text{Tran}(G_h^u) \quad (4)$$

where G_h^u is the gradient magnitudes of ∇I_h^u . Accordingly, the finally sharpened gradient field $\widehat{\nabla I_h}$ is obtained by

$$\widehat{\nabla I_h} = \widehat{G}_h \cdot \theta_h^u. \quad (5)$$

A. Scaling Sharpening Overview

In [5] and [29], they define the transformation function as follows (Ω is the image domain):

$$\widehat{G}_h(\mathbf{x}) = \text{Tran}(G_h^u) = r(\mathbf{x})G_h^u(\mathbf{x}) \quad \mathbf{x} \in \Omega \quad (6)$$

where ratio r is learnt from natural images. Yan *et al.* [30] improve the ratio calculation by using the Laplacian image. As interpreted in [5], [29], and [30], the main purpose of

multiplying ratio r is to sharpen the gradient magnitude, especially the gradient profile [see Fig. 2(d) and (e)]. For example, in Fig. 2(f), the ratio r along the gradient profile should satisfy

$$\begin{cases} r(x_Y) \geq 1 & x_Y \in [y_2 \ y_3] \\ 0 \leq r(x_Y) < 1 & x_Y \in [y_1 \ y_2] \cup (y_3 \ y_4] \end{cases} \quad (7)$$

where x_Y is the 1-D coordinate value along the y -axis. The ratio r proposed in [29] used to zoom the gradient magnitude; thereby, we call this gradient sharpening method scaling sharpening.

B. Proposed Self-Interpolation Sharpening

We propose a self-interpolation sharpening to sharpen the gradient magnitude. We use an interpolation operator instead of a zoom operator to make the gradient magnitude sharp. That is, the sharp gradient magnitude \widehat{G}_h is obtained through the interpolation function

$$\widehat{G}_h = \text{Tran}(G_h^u) = \text{Intp}(G_h^u, \mathbf{u}) \quad (8)$$

where \mathbf{u} is the displacement field and $\text{Intp}(\cdot)$ is the interpolation function defined as follows:

$$\widehat{G}_h(\mathbf{x}) = \text{Intp}(G_h^u(\mathbf{x}), \mathbf{u}(\mathbf{x})) = G_h^u(\mathbf{x} - \mathbf{u}(\mathbf{x})) \quad \mathbf{x} \in \Omega. \quad (9)$$

Fig. 2(g) gives a 1-D interpolation example. As shown in this subfigure, the green arrows illustrate the displacement field \mathbf{u} . The interpolated gradient magnitude (see the blue curve) is much sharper than the original gradient magnitude (see the red curve), after performing interpolating operation.

On the other hand, as shown in Fig. 2(g), to make G_h^u sharp by the interpolation method, the displacement field \mathbf{u} should satisfy that

$$\begin{cases} \mathbf{u}(x_Y) > 0 & x_Y < y_o \\ \mathbf{u}(x_Y) = 0 & x_Y = y_o \\ \mathbf{u}(x_Y) < 0 & x_Y > y_o \end{cases} \quad (10)$$

where y_o is the local peak point of the gradient profile. Fortunately, the gradient of original gradient magnitude, i.e., ∇G_h^u , just satisfies (10), that is

$$\begin{cases} \nabla G_h^u(x_Y) > 0 & x_Y < y_o \\ \nabla G_h^u(x_Y) = 0 & x_Y = y_o \\ \nabla G_h^u(x_Y) < 0 & x_Y > y_o. \end{cases} \quad (11)$$

The pink curve of Fig. 2(g) shows ∇G_h^u . From this subfigure, we conclude that the sign of the displacement field \mathbf{u} is the same as the sign of ∇G_h^u . Thus, we can efficiently estimate the displacement field \mathbf{u} from ∇G_h^u directly.

Based on the above analysis, the estimation of the displacement field \mathbf{u} only relies on the original gradient magnitude. Thereby, we call our gradient sharpening method self-interpolation sharpening. More information about gradient magnitude self-interpolation will be presented in Section V in detail.

In the following, we use a 1-D case to justify that our self-interpolation-based method can sharpen the gradient magnitude (the gradient profile is 1-D). Let $f(x)$ be a continuous and

monotonic function defined in (a, b) ; the sharpness of $f(x)$ in (a, b) is denoted as

$$\omega_{(a,b)}(f(x)) = \frac{\int_a^b f(x) dx}{|f(b) - f(a)|}. \quad (12)$$

The sharpness $\omega_{(a,b)}(f(x))$ is interpreted as the mean width of $f(x)$ [refer to the rectangle width of Fig. 3(a)]. The sharper the function $f(x)$, the smaller the $\omega_{(a,b)}(f(x))$.

Definition 1: $f(x)$ and $g(x)$ are both continuous and monotonic functions defined in (a, b) , satisfying $f(a) = g(a)$ and $f(b) = g(b)$.

- 1) If $f(x)$ and $g(x)$ are both monotonically increasing, and $f(x) \geq g(x)$, $\forall x \in (a, b)$, then $f(x)$ is smoother than $g(x)$, i.e., $\omega_{(a,b)}(f(x)) \geq \omega_{(a,b)}(g(x))$.
- 2) If $f(x)$ and $g(x)$ are both monotonically decreasing, and $f(x) \geq g(x)$, $\forall x \in (a, b)$ then $f(x)$ is smoother than $g(x)$, i.e., $\omega_{(a,b)}(f(x)) \geq \omega_{(a,b)}(g(x))$.

Definition 1 gives a criterion to compare the sharpness of two functions. The intuitive comparisons are illustrated in Fig. 3(b) and (c) in detail.

Lemma 1: $f(x)$ is a continuous and monotonically increasing function defined in (a, b) . The derivative $f'(x)$ is a bounded and continuous function, satisfying $f'(a) = f'(b) = 0$. Defining $g(x) = f(x - \eta f'(x))$, where $\eta > 0$ and $1 - \eta f''(x) \geq 0$, we get that

$$\omega_{(a,b)}(f(x)) \geq \omega_{(a,b)}(g(x)).$$

For a monotonically decreasing function $f(x)$, we can get a similarity result.

Proof: First, according to $f'(a) = f'(b) = 0$, we get that $g(a) = f(a + f'(a)) = f(a)$, $g(b) = f(b + f'(b)) = f(b)$. $f'(x)$ is bounded and continuous; thus, $g(x) = f(x - \eta f'(x))$ is also a continuous function. Let $t(x) = x - \eta f'(x)$. Thereby, $t'(x) = 1 - \eta f''(x) \geq 0$. Hence, $\forall a \leq x_1 \leq x_2 \leq b$. We get that $g(x_1) = f(t(x_1)) \leq f(t(x_2)) = g(x_2)$. To sum up, $g(x)$ is a continuous and monotonically increasing function.

Second, $f(x)$ is a continuous and monotonically increasing function; thus, $f'(x) \geq 0$, which makes $x \geq x - \eta f'(x)$. Hence, $g(x) = f(x - \eta f'(x)) \leq f(x)$, $\forall x \in (a, b)$.

Combining with Definition 1, we get $\omega_{(a,b)}(f(x)) \geq \omega_{(a,b)}(g(x))$. ■

Comparing (9) with Lemma 1, $\eta f'(x)$ can be regarded as a displacement field in the interpolation function. It is worth noting that $\eta f'(x)$ is obtained from its gradient. Hence, for a continuous and monotonically function, self-interpolation can sharpen the gradient theoretically. In practice, the gradient profile function can be assumed to be composed of some continuous and monotonically functions. For example, the gradient profile can be approximated with a Gaussian function. In such a case, the definition domain of $f(x)$ can be divided into two parts, i.e., $(-\infty, 0)$ and $(0, \infty)$. It is easy to check that in each definition domain, $f(x)$ satisfies Lemma 1.

V. ADAPTIVE GRADIENT MAGNITUDE SELF-INTERPOLATION

As interpreted in Section III, the key problem of our self-interpolation sharpening is the estimation of displacement

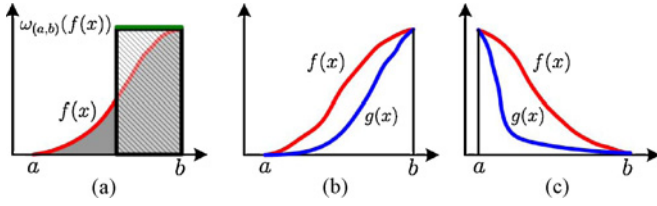


Fig. 3. (a) Sharpness $\omega_{(a,b)}(f(x))$ is defined as the mean width of $f(x)$, or the rectangle width (green line). (b), (c) Intuitive comparisons of two functions.

field \mathbf{u} . The input gradient magnitude G_h^u is 2-D. Therefore, the corresponding displacement field \mathbf{u} should also have two directions, namely, $\mathbf{u} = (u_x, u_y)$, where u_x and u_y are the x - and y -direction displacement fields, respectively. Similarly to the definition used for the gradient field, the displacement field \mathbf{u} is also denoted by the dot product of its magnitude M and direction ϕ , given by

$$\mathbf{u} = (u_x, u_y) = M \cdot \phi. \quad (13)$$

In the following, we first present the estimation of direction ϕ and magnitude M , and then summarize the main steps of our gradient sharpening algorithm.

A. Direction ϕ Estimation

As mentioned in Section III, in the 1-D case, the sign of the displacement field just equals that of the original gradient magnitude [refer to (10) and (11) for details]. It is easy to extent the 1-D case to the 2-D case by replacing the sign with the angle between two directions. Hence, the direction ϕ can be computed as follows:

$$\phi = \arctan \left(\frac{\partial_y G_h^u}{\partial_x G_h^u} \right). \quad (14)$$

In practice, the direction calculated by (14) is a little sensitive to the noise. Hence, the direction is further smoothed by Gaussian kernel g , namely, $\phi = \phi \otimes g$. During implementation, the standard variance of g is set to 3.

B. Estimation of the Magnitude M

From Fig. 2(g), in a gradient profile, the magnitude of displacement (length of green arrows) gradually becomes larger, as the distance between the current coordinate value and the peak point becomes larger, that is

$$|y_i - y_o| > |y_j - y_o| \Rightarrow M(y_i) > M(y_j) \quad (15)$$

where y_o is the local peak point of original gradient magnitude. As illustrated in Fig. 2(g), the gradient magnitude of G_h^u satisfies that (see the pink curve)

$$\begin{cases} |y_i - y_o| < |y_j - y_o| \Rightarrow |\nabla G_h^u(y_i)| > |\nabla G_h^u(y_j)|, & y_1 \leq y_i, y_j < y_2 \\ |y_i - y_o| > |y_j - y_o| \Rightarrow |\nabla G_h^u(y_i)| > |\nabla G_h^u(y_j)|, & y_2 \leq y_i, y_j \leq y_3 \\ |y_i - y_o| < |y_j - y_o| \Rightarrow |\nabla G_h^u(y_i)| > |\nabla G_h^u(y_j)|, & y_3 < y_i, y_j \leq y_4 \end{cases} \quad (16)$$

From (15) and (16), we see that $|\nabla G_h^u|$ and M share the same condition when $y_2 \leq y_i, y_j \leq y_3$. Furthermore, the error between $M(y)$ and $|\nabla G_h^u(y)|$ becomes larger as the coordinate y becomes closer to the two edge points y_1 or y_4 . Fortunately, around y_1 or y_4 , the change of gradient

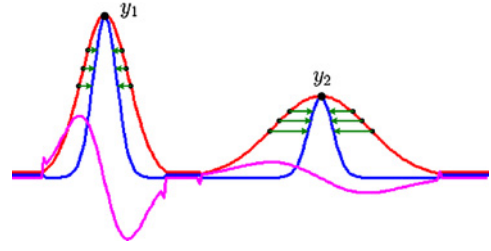


Fig. 4. Example of multiple scales gradient magnitude sharpening. The red curve is the original gradient magnitude, the blue curve is the sharpened gradient magnitude, and the pink curve is the gradient of original gradient magnitude.

magnitude of G_h^u is very small. That is, the interpolation result obtained by (9) almost does not change even the estimation of displacement field (either magnitude or direction), and has obvious errors around y_1 or y_4 . Hence, $|\nabla G_h^u|$ can be adopted as the approximate estimation of the magnitude M . In the 2-D case, M is calculated as follows:

$$M = \kappa \sqrt{(\partial_x G_h^u)^2 + (\partial_y G_h^u)^2} \quad (17)$$

where κ is a scaling factor. The choice of κ is mainly related to the up-sampling factors and the characters of the input LR image. The κ are experimentally set to ≈ 0.6 , ≈ 0.8 , and ≈ 1 for the up-sampling factors of 2, 3, and 4.

As shown in (17), the parameter κ governs the scaling value globally. In practical, this global method cannot deal with the multiple scales case. Fig. 4 gives an example of multiple scales gradient magnitude sharpening. As shown in this figure, the original gradient magnitude (red curve) around y_1 is sharper than that around y_2 . Hence, to sharpen the gradient, the displacement magnitudes around y_1 should be smaller than those around y_2 . However, the gradient of original gradient magnitude $|\nabla G_h^u|$ (see the pink curve) just provides the opposite result, i.e., $|\nabla G_h^u|$ around y_1 is larger than those around y_2 . To solve multiple scales case, we should adaptively calculate the scaling factor $\kappa(\mathbf{x})$ according to the local scale of gradient profile

$$\kappa(\mathbf{x}) = \sigma(\mathbf{x})\kappa \quad \mathbf{x} \in \Omega \quad (18)$$

where $\sigma(\mathbf{x})$ is the local scale located \mathbf{x} . The corresponding magnitude M is calculated as follows:

$$M(\mathbf{x}) = \kappa(\mathbf{x}) \sqrt{(\partial_x G_h^u)^2 + (\partial_y G_h^u)^2} \quad \mathbf{x} \in \Omega. \quad (19)$$

In this paper, the local scale σ is obtained by two substeps: 1) ridge points (all points on ridge lines) detection, and local scale estimation of these ridge points, and 2) local scale value filling.

In the first step, we perform the ridge lines detection and width estimation algorithm proposed in [31] on the gradient magnitude to obtain the ridge points $p(\mathbf{x})$ and their corresponding width $w(\mathbf{x})$, given by

$$\{p(\mathbf{x}), w(\mathbf{x}) | \mathbf{x} \in \mathcal{X}\}$$

where $\mathcal{X} \subset \Omega$ is a point position set of all ridge points. The local scales of these ridge points are calculated by

$$\sigma(\mathbf{x}) = \max \left(\xi_l, \min \left(\xi_h, \frac{w(\mathbf{x})}{\text{med}(w)} \right) \right) \quad \mathbf{x} \in \mathcal{X} \quad (20)$$

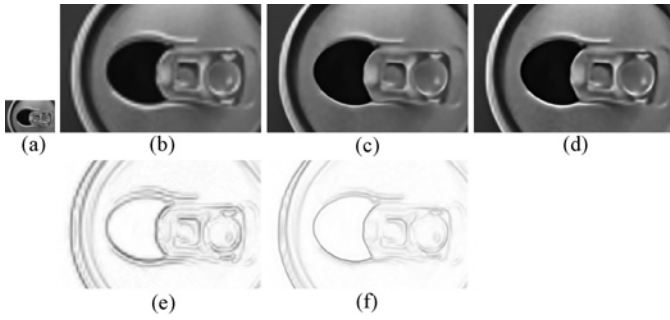


Fig. 5. Super-resolution example (4X magnification). (a) Input LR image. (b) Bicubic. (c) Initialization of our method. (d) Our result. (e) Gradient maps (normalized and inverted magnitudes) of (b). (f) Our adaptively self-interpolated gradient maps.

Algorithm 1: Adaptive Gradient Sharpening

Data: The bi-cubic up-sampled gradient ∇I_h^u

Result: The sharpened gradient $\widehat{\nabla I_h}$

- 1 Calculate original gradient magnitude and direction by $\nabla I_h^u = G_h^u \cdot \theta_h^u$;
 - 2 Calculate displacement field direction ϕ by Eqn. (14);
 - 3 Calculate displacement field magnitude M by Eqns (20), (18) and (19);
 - 4 Calculate displacement field by Eqn. (13);
 - 5 Interpolate the original gradient magnitude to obtain the sharpened gradient magnitude by Eqn. (9);
 - 6 Output the estimated sharpened gradient field by Eqn. (5);
-

where ξ_l and ξ_h are the low and high bounds, respectively, and $\text{med}(w)$ gives the median width of all ridge points. During implementation, ξ_l and ξ_h are set $\xi_l = 0.5$ and $\xi_h = 2$.

In the second step, we fill the rest points by the nearest neighboring filling methods, namely

$$\sigma(\mathbf{y}) = \sigma(\mathbf{x}), \quad \mathbf{y} \in \Omega - \mathcal{X} \quad (21)$$

where $p(\mathbf{x})$ is the nearest neighbor of $p(\mathbf{y})$ in the set \mathcal{X} .

C. Algorithm

To sum up, the main procedures of our adaptive gradient sharpening method are summarized in Algorithm 1. Fig. 5 gives an example of our super-resolution result. From this figure, the adaptively self-interpolated gradient magnitude [see Fig. 5(f)] is significantly sharper than the original one [see Fig. 5(e)], especially on the salient edges.

VI. OPTIMIZATION AND INITIAL HR IMAGE SETTING

The objective energy functional of (2) is minimized by the standard gradient descent by solving the gradient flow equation given by

$$\begin{aligned} \frac{\partial I_h}{\partial t} &= - \frac{\partial E(I_h | I_l, \widehat{\nabla I_h})}{\partial I_h} \\ &= - \frac{\partial \left(\| [I_h \otimes g]_{\downarrow(\beta)} - I_l \|^2 + \alpha \|\nabla I_h - \widehat{\nabla I_h}\|^2 \right)}{\partial I_h} \\ &= - \left([I_h \otimes g]_{\downarrow(\beta)} - I_l \right)_{\uparrow(1/\beta)} \otimes g + \\ &\quad \alpha \left(\text{div}(\nabla I_h) - \text{div}(\widehat{\nabla I_h}) \right) \end{aligned} \quad (22)$$

where $\text{div}(\nabla \cdot) = \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2}$ is the Laplacian operator. In our implementation, we use the following iterative scheme to optimize (22), that is

$$\begin{aligned} I_h^{t+\tau} &= I_h^t - \tau \left(([I_h \otimes g]_{\downarrow(\beta)} - I_l)_{\uparrow(1/\beta)} \right) \otimes g - \\ &\quad \alpha \left(\text{div}(\nabla I_h) - \text{div}(\widehat{\nabla I_h}) \right) \end{aligned} \quad (23)$$

where τ is time step. During implementation, the time step τ is set to 0.1. The parameter α is used to trade off between the two constraints, i.e., an image domain constraint and a gradient domain constraint. A smaller α will produce better image color and contrast, yet with ringing or jaggy artifacts along edges. On the contrary, a larger α will result in sharp edges with little artifacts. To balance the image color recovery and artifacts removal, the parameter α is experimentally set to 0.1.

As described above, an iteration method is used to solve (22); thereby, the super-resolution result partially depends on the initial HR image. Generally, the bicubic up-sampled image I_h^u is regarded as the initial HR image I_h^{init} , namely, $I_h^{\text{init}} = I_h^u$. However, the up-sampled image I_h^u is blurred in the image edges, which causes that the super-resolution algorithm needs many iterations to obtain the sharp edge, for example, 100 iterations reported in [29]. Hence, the computational cost will be high. To avoid this, we use the interpolated version as the initialization given by

$$I_h^{\text{init}} = \text{Intp}(I_h^u, \mathbf{u}) \quad (24)$$

where $\text{Intp}(\cdot)$ is defined in (9). Fig. 5 gives an example. As shown in Fig. 5(c), the initialization obtained by our method is much more sharper than the bicubic [see Fig. 5(b)]. In our method, 30 iterations are enough to produce sharp and clear HR results.

VII. EXPERIMENTAL RESULTS

Extensive experiments have been conducted to evaluate our method in comparison with several state-of-the-arts, which cover the four categories of single-image super-resolution, i.e., interpolation-based, reconstruction-based, learning-based, and edge-directed methods. Note that, for a color image, it is first transformed from RGB to YIQ. Then, the Y channel (intensity) is up-sampled by our algorithm. The I and Q chromatic channels, characterized by low-frequency information, are interpolated by the bicubic method. Finally, the three channels are combined to form the final super-resolution result. Both qualitative and quantitative methods are utilized to evaluate our method. For quantitative evaluation, we use RMS, ERMS [16], [29], and SSIM [32] to measure super-resolution results. A good super-resolution result should provide small RMS and ERMS, and large SSIM.

A. Evaluation of the Proposed Method

In this subsection, we evaluate the initialization and key parameter of our method. For initialization, we compare our method with the GPP edge-directed method [29].

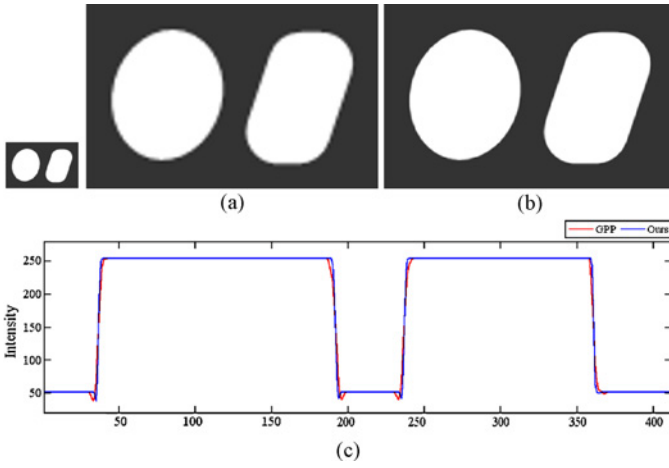


Fig. 6. Initialization comparison on synthetic image (4X magnification). (a) Bicubic initialization of GPP. (b) Our initialization. (c) Comparison of the edge sharpness. To compare sharpness, for each image, we select the pixel values in the 140th row.

TABLE I
INITIALIZATION COMPARISON ON 300 IMAGES OBTAINED FROM
BERKELEY DATABASE

| Mag. & Criteria | Method | GPP | Our Method |
|-----------------|--------|-------|------------|
| | | | |
| X3 | RMS | 13.48 | 13.60 |
| | ERMS | 21.49 | 21.00 |
| | SSIM | 0.738 | 0.739 |
| X4 | RMS | 15.51 | 15.40 |
| | ERMS | 24.97 | 23.91 |
| | SSIM | 0.665 | 0.670 |

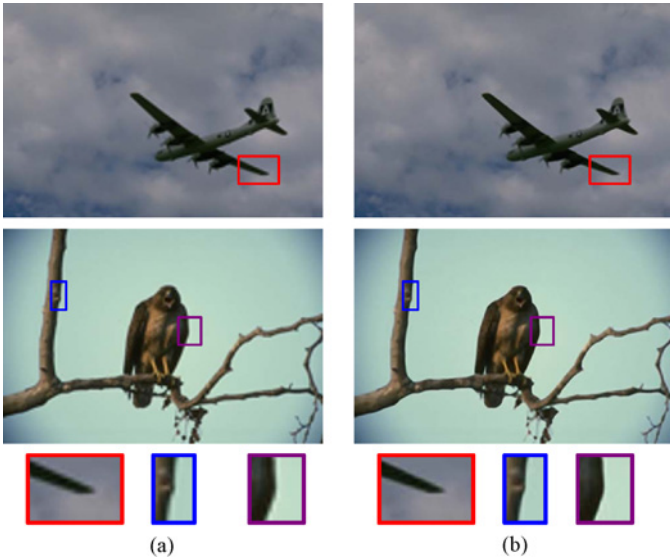


Fig. 7. Initialization comparison on two real images (4X magnification). (a) Bicubic initialization of GPP. (b) Our initialization.

1) *Initialization Evaluation*: Fig. 6 first presents a synthetic comparative result. As shown in this figure, our initialization image is sharper than that of GPP edge-directed. For further comparison, we test two methods on 300 images, which are obtained from the Berkeley website [33]. The original image is used as the ground truth. The LR image

TABLE II
EFFECT OF PARAMETER α ON 300 IMAGES SUPER-RESOLUTION RESULTS

| Parameter | $\alpha = 0.05$ | $\alpha = 0.075$ | $\alpha = 0.1$ | $\alpha = 0.125$ | $\alpha = 0.15$ | $\alpha = 0.2$ | $\alpha = 0.25$ |
|-----------|-----------------|------------------|----------------|------------------|-----------------|----------------|-----------------|
| Criteria | | | | | | | |
| RMS | 12.84 | 12.84 | 12.86 | 12.84 | 12.85 | 12.85 | 12.85 |
| ERMS | 18.76 | 18.78 | 18.67 | 18.82 | 18.84 | 18.89 | 18.94 |
| SSIM | 0.767 | 0.767 | 0.768 | 0.767 | 0.766 | 0.766 | 0.765 |

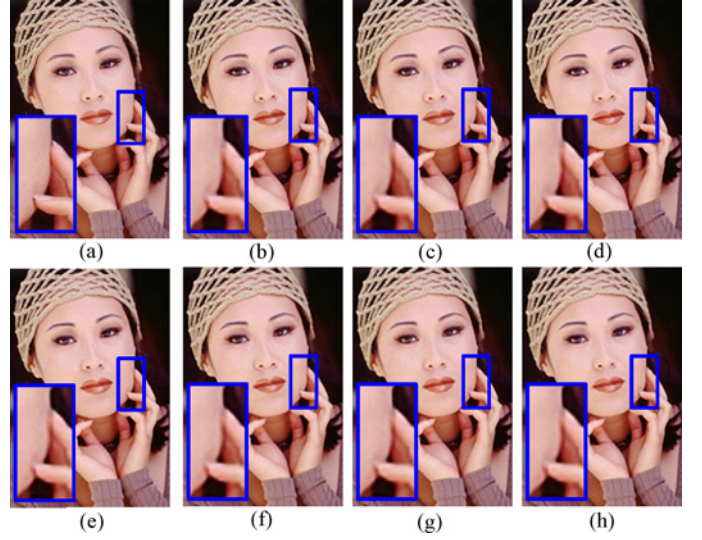


Fig. 8. Effect of parameter α on the super-resolution results. (a) Original image. (b)–(h) are the up-sampled results (3X) with $\alpha = 0.05, 0.075, 0.1, 0.125, 0.15, 0.2, 0.25$.

is obtained by down sampling its original version. The quantitative evaluations are shown in Table I. From this table, our results are better than those of GPP edge-directed, especially when the up-sampling factor is higher. Fig. 7 presents two examples from the 300 images. From this figure, we see that our initializations are sharper.

2) *Parameter α Evaluation*: We test our algorithm with different α on 300 image obtained from Berkeley database. The ground truth and LR images are obtained in the same way as given in Section VII-A1. The quantitative results are shown in Table II. The visual results are illustrated in Fig. 8. From Table II and Fig. 8, we see that our results are very stable under different choices of α . Moreover, when $\alpha = 0.1$, the ERMS and SSIM are the best. In other experiments, the parameter α is set 0.1.

B. Comparisons of Other Edge-Directed Methods

In this subsection, we compare our method with three edge-directed methods, i.e., the GPP edge-directed [29],¹ the soft-cut edge-directed [11],² and the Laplacian edge-directed [30].³

¹The results are obtained from yuw.kaist.ac.kr/projects/superresolution/index.htm. In this website, Tai *et al.* [5] provide the code about GPP implementation.

²The results are available at vision.eecs.northwestern.edu/research/IP/SR/index.html.

³We implement the Laplacian edge-directed super-resolution algorithm according to their paper [30], and some parameters are adjusted according to the input image.

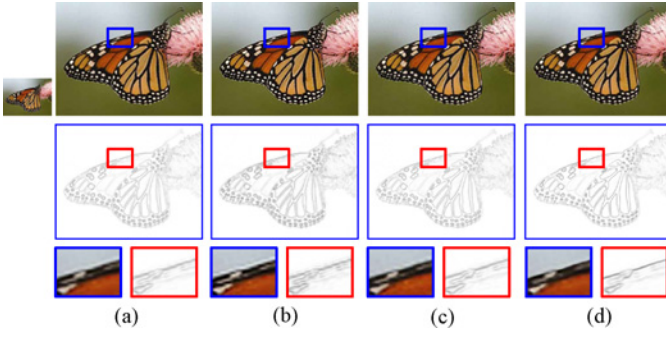


Fig. 9. Super-resolution comparison (3X magnification) of other edge-directed methods. (a) Back-projection. (b) Laplacian edge-directed [30]. (c) GPP edge-directed [29]. (d) Our result. The second row presents the gradient magnitude (normalized and inverted magnitude) detail of up-sampled images. The third row illustrates the closeups. Please refer to the electronic version for a better comparison.

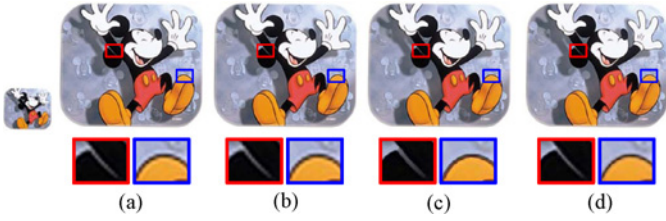


Fig. 10. Super-resolution comparison (3X magnification) of other edge-directed methods. (a) Back-projection. (b) Laplacian edge-directed [30]. (c) Soft-cut edge-directed [11]. (d) Our result. The bottom row illustrates the closeups of the corresponding results.

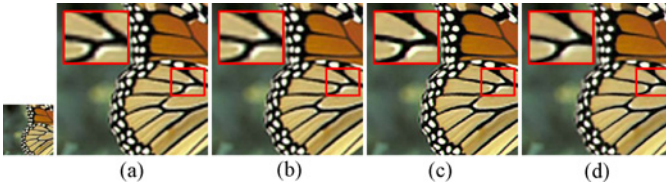


Fig. 11. Super-resolution comparison (3X magnification) of other edge-directed methods. (a) Back-projection. (b) Laplacian edge-directed [30]. (c) Soft-cut edge-directed [11]. (d) Our result.

1) *Comparison With GPP [29]*: Fig. 9 illustrates the comparison of our method with the GPP edge-directed and Laplacian edge-directed methods in detail. As shown in this figure, the edges are significantly blurred by the back-projection and Laplacian edge-directed methods (see closeups). The ringing artifacts along the image edges of our method are less than those of the GPP edge-directed method. Particularly, this figure also shows that the gradient magnitude of our method is sharper than those of other methods, especially along the edges (see closeups).

2) *Comparison With Soft-Cut [11]*: Figs. 10 and 11 present two comparisons of our approach with the soft-cut edge-directed and Laplacian edge-directed methods. As with the results in Fig. 9, the image edges are blurred by the back-projection and Laplacian edge-directed methods. Moreover, these results also have obviously ringing artifacts along edges (refer to close-ups). In the soft-cut edge-directed super-resolution results, some salient edges are blocked and look unnatural. For example, as shown in Fig. 11(c), the blocking

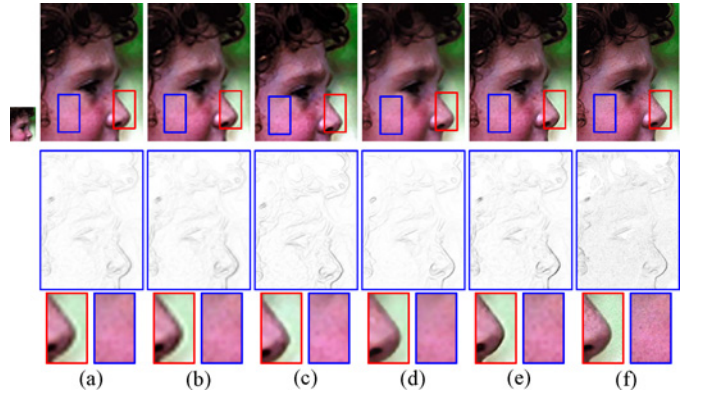


Fig. 12. Super-resolution comparison (4X magnification) of other edge-directed methods. (a) Back-projection. (b) Laplacian edge-directed [30]. (c) GPP edge-directed [29]. (d) Soft-cut edge-directed [11]. (e) Our result. (f) Ground truth. The second row presents the gradient magnitude (normalized and inverted magnitude) detail of up-sampled images.

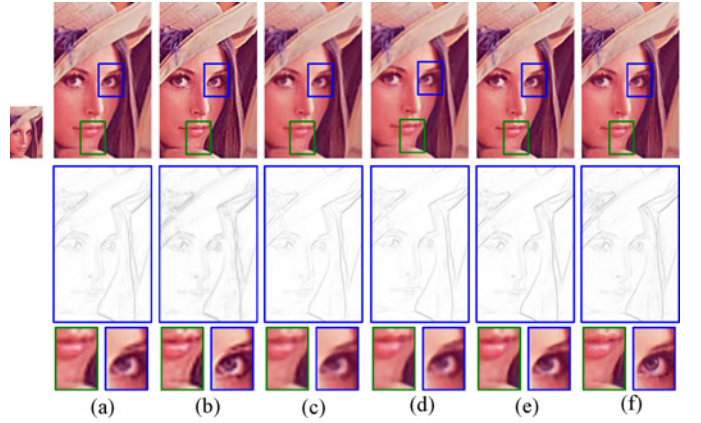


Fig. 13. Super-resolution comparison (3X magnification) of other edge-directed methods. (a) Back-projection. (b) Laplacian edge-directed [30]. (c) GPP edge-directed [29]. (d) Soft-cut edge-directed [11]. (e) Our result. (f) Ground truth. The second row presents the gradient magnitude (normalized and inverted magnitude) detail of up-sampled images.

artifacts are obvious along both sides of the black dapple of the butterfly. Compared with these results, our approach recovers the super-resolution results better, especially on salient edges.

3) *Comparison With GPP [29] and Soft-Cut [11]*: Figs. 12 and 13 present two more super-resolution examples by comparing with GPP, soft-cut, and Laplacian edge-directed methods. As shown in these two figures, the back-projection and Laplacian edge-directed algorithms produce serious ringing artifacts, such as the edges around nose in Fig. 12. The salient edges in the soft-cut results are very sharp, but the small scale edges are not well recovered, for example, the flecks on the face in Fig. 12, and the texture areas on the hat in Fig. 13. The main reason is that in the soft-cut edge-directed method, the alpha channel in a small scale area is very hard to estimate. Hence, the gradient field obtained by this method is very smooth, which further causes the loss of details. From the second rows of these two figures, the gradient magnitudes of GPP edge-directed are less sharper than those obtained through our method and the soft-cut method. In particular, both the figures show that the gradient magnitudes of our method are

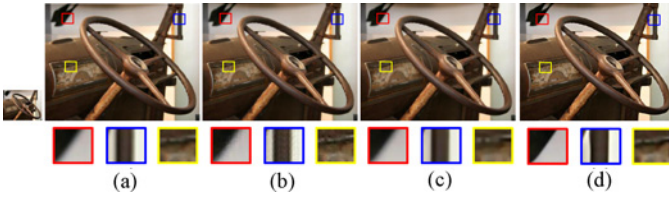


Fig. 14. Super-resolution comparison (4X magnification) of learning based methods. (a) Bicubic. (b) Yang *et al.*'s [7]. (c) Shan *et al.*'s [34]. (d) Our result. The running time of our method is about 20 s. Yang *et al.*'s needs 80 s.

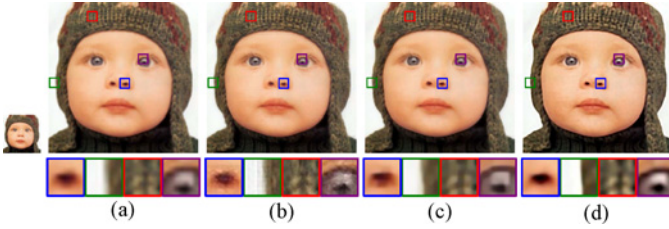


Fig. 15. Super-resolution comparison (4X magnification) of learning based methods. (a) Bicubic. (b) Yang *et al.*'s [7]. (c) Shan *et al.*'s [34]. (d) Our result. The running time of our method is about 14 s. Yang *et al.*'s needs 50 s.

more similar to the ground truth than those of GPP and soft-cut methods. This indicates that the gradient fields obtained through our method are more accurate.

C. Comparisons With Learning and De-convolution Based Methods

In this subsection, we compare the proposed method with four methods, i.e., Yang *et al.*'s [7],⁴ Shan *et al.*'s [34] (de-convolution based),⁵ Glasner *et al.*'s [6],⁶ and Freedman *et al.*'s [24].⁷

1) *Comparison With Yang et al.'s [7] and Shan et al.'s [34]*: In the first example, we compare our method with Yang *et al.*'s and Shan *et al.*'s in two aspects, i.e., super-resolution results and computational cost. In our method, the iteration times are equally set 30. As can be seen from Figs. 14 and 15, the results of Yang *et al.*'s are sharper in appearance than those by the bicubic (see close-ups for detail comparisons). However, the high-frequency artifacts are also introduced from the training samples, such as the ringing artifacts around the pipelines. In Shan *et al.*'s [34] results, the salient edges are smoother than those of our method [see the comparison of red patches in Fig. 14(c), (d) as an example]. The speed of our method is faster. For example, in Fig. 14, the running time of our method is about 20 s. Yang *et al.*'s needs 80 s. It is worth noting that we run all algorithms on PC with CPU as a processor. Compared with Yang *et al.*'s and Shan *et al.*'s methods, our edge-directed method not only recovers the HR image with less visual artifacts, but also takes a lower computational cost.

⁴Code available at www.ifp.illinois.edu/~jyang29/.

⁵The executable code is downloaded from www.cse.cuhk.edu.hk/~leojia/projects/upsampling/index.html.

⁶The results are available at www.wisdom.weizmann.ac.il/~vision/SingleImageSR.html.

⁷The results are available at www.cs.huji.ac.il/~raananf/projects/lss_upscale/index.html.

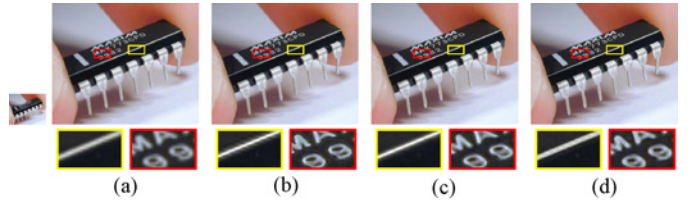


Fig. 16. Super-resolution comparison (4X magnification) of learning based methods. (a) Bicubic. (b) Glasner *et al.*'s [6]. (c) Freedman *et al.*'s [24]. (d) Our result. The bottom rows illustrate the close-ups of the corresponding results.

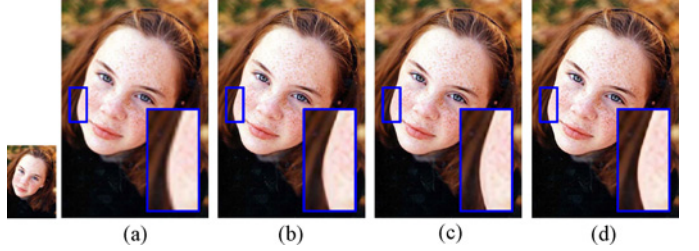


Fig. 17. Super-resolution comparison (3X magnification) of learning based methods. (a) Bicubic. (b) Glasner *et al.*'s [6]. (c) Freedman *et al.*'s [24]. (d) Our result.

2) *Comparison With Glasner et al.'s [6] and Freedman et al.'s [24]*: Figs. 16 and 17 present two examples of comparing with Glasner *et al.*'s and Freedman *et al.*'s methods. In total, as shown in these two figures, the details recovered by our method are better than those by Freedman *et al.*'s method, and the ringing artifacts introduced by our method are less than those by the Glasner *et al.*'s method. Take Fig. 16 as an example, the edges on the letters A and 9 (see red close-ups) are too sharp to look natural in Freedman *et al.*'s result. On the other hand, the edges in Glasner *et al.*'s method are not well recovered (see yellow close-ups). Our method can reconstruct sharp edges, but with a little additional artifact.

To sum up, these comparisons show that our method is able to generate sharper edges while producing less artifacts than the learning-based super-resolution methods. Meanwhile, the computational cost of our method is lower. Furthermore, as a general method on the image gradient, our gradient sharpening algorithm can also be applied to other image processing tasks, such as image enhancement and image de-focus.

D. Comparison on Dataset

Table III gives the numeric comparisons of our method with the bicubic, back-projection [25], Laplacian edge-directed [30], and Yang *et al.*'s [7] methods. The tested images are obtained from the Berkeley website [33]. Each image has the corresponding edge mask. Three up-sampling scales are adopted to make the comparison more thoroughly. The original image is used as the ground truth. The LR image is obtained by down sampling its original version. The EMS, ERMS, and SSIM error metrics are used to quantitatively measure the super-resolution results. We calculate the average errors of all tested images. As shown in the Table III, our method presents lower EMS and ERMS and higher SSIM, as compared with others, especially when the up-sampling scale is larger. Furthermore, the computational cost of our method is significantly

TABLE III
SUPER-RESOLUTION QUALITY MEASUREMENT ON DATA-SET

| Method Meg. & Criteria | Bi-cubic | Back- projection | Learning | Laplacian | Our Method |
|------------------------------|----------|---------------------|--------------|-----------|---------------|
| 2X | RMS | 10.30 | 10.41 | 12.79 | 10.34 |
| | ERMS | 16.08 | 15.04 | 17.82 | 14.71 |
| | SSIM | 0.844 | 0.856 | 0.835 | <u>0.850</u> |
| | CPU Time | 0.080 | 4.786 | 77.16 | 5.067 |
| 3X | RMS | 13.48 | 13.05 | 14.47 | 13.14 |
| | ERMS | 21.49 | 19.79 | 20.90 | <u>19.29</u> |
| | SSIM | 0.738 | 0.763 | 0.757 | <u>0.756</u> |
| | CPU Time | 0.062 | 3.724 | 38.57 | 3.917 |
| 4X | RMS | 15.51 | 14.95 | 15.97 | 14.93 |
| | ERMS | 24.97 | 23.39 | 25.05 | <u>23.39</u> |
| | SSIM | 0.665 | 0.690 | 0.646 | 0.689 |
| | CPU Time | 0.055 | 3.385 | 24.04 | 3.828 |

We compared our algorithm with the bi-cubic, back-projection, Yang et al.'s [7], and Laplacian edge-directed [30] algorithms by three error measurements, i.e., RMS, ERMS, and SSIM. The CPU time rows present the running time (s). (**Bold**: best, underline: second best).

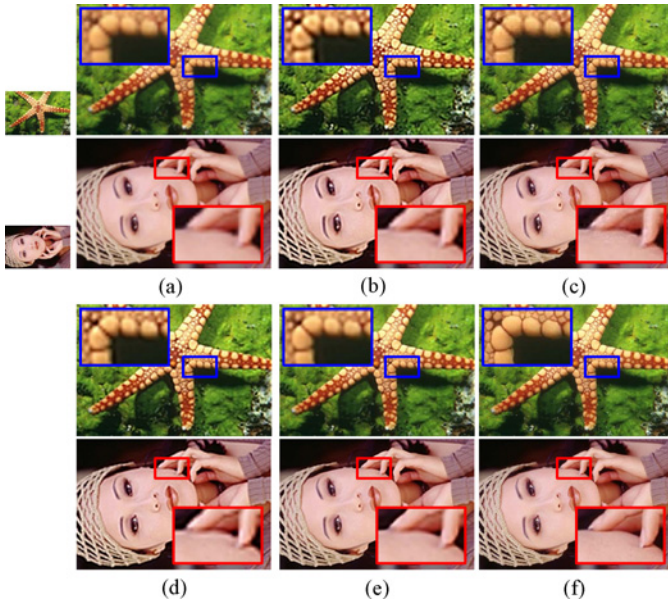


Fig. 18. Super-resolution comparison (3X magnification) of other methods. (a) Bicubic. (b) Back-projection. (c) Yang et al.'s [7]. (d) Laplacian edge-directed [30]. (e) Our result. (f) Ground truth.

TABLE IV
SUPER-RESOLUTION QUALITY MEASUREMENT ON TWO EXAMPLES

| Method Mag. & Criteria | Bi-cubic | Back- projection | Learning | Laplacian | GPP | Our Method |
|---------------------------|----------|---------------------|--------------|-----------|--------------|---------------|
| Star Fish | RMS | 11.62 | 11.19 | 13.46 | <u>11.36</u> | 11.5 |
| | ERMS | 15.50 | 13.42 | 15.82 | 13.56 | 12.0 |
| | SSIM | 0.798 | <u>0.823</u> | 0.807 | 0.812 | <u>0.790</u> |
| Lady | RMS | 10.09 | <u>9.627</u> | 12.18 | 9.505 | 9.5 |
| | ERMS | 17.56 | 14.54 | 15.68 | 13.86 | <u>12.3</u> |
| | SSIM | 0.886 | 0.889 | 0.858 | 0.878 | <u>0.891</u> |

We compared our algorithm with the bi-cubic, back-projection, Yang et al.'s [7], Laplacian edge-directed [30], and GPP edge-directed [29] algorithms by three error measurements, i.e., RMS, ERMS, and SSIM. The CPU time rows present the running time (s). (**Bold**: best, underline: second best).

lower than that of Yang *et al.*'s method. Note that the iteration times for back-projection, Laplacian edge-directed, and our method are equally set to 30.

We select two images, i.e., *Star Fish* and *Lady*, as examples for detailed comparison. The visual comparison results are presented in Fig. 18, while the quantitative comparison results are shown in Table IV. The quantitative results of GPP are obtained from their paper [29]. As shown in this table, our method is better than the others. Moreover, as shown in Fig. 18, our method recovers sharper HR images. As reported in [29], the gradient sharpening times for these two images are 0.829 and 0.862 s, respectively. However, in our method, it only needs 0.266 and 0.269 s to sharpen these gradient. Hence, the gradient sharpening time of our method is greatly lower (three times) that those of [29].

VIII. CONCLUSION AND FUTURE WORK

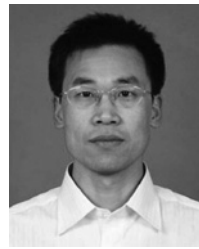
In this paper, an edge-directed single-image super-resolution algorithm was proposed by using a new adaptive gradient magnitude self-interpolation. The extensive experimental results demonstrated that the proposed gradient constraint, which is represented by the estimated HR gradient, can preserve image details or sharp edges, while suppressing the ringing, blocking, and blurring artifacts, especially along salient edges.

In the future, we want to speed up the proposed super-resolution algorithm. It is also interesting to incorporate the learning-based models to enhance the super-resolution results. In addition to the super-resolution problem, the proposed gradient constraint can also be effectively applied to other computer graphic and computer vision problems, such as image de-blurring, de-noising, and enhancement.

REFERENCES

- [1] S. Farsiu, D. Robinson, M. Elad, and P. Milanfar, "Fast and robust multiframe super resolution," *IEEE Trans. Image Process.*, vol. 13, no. 10, pp. 1327–1344, Oct. 2004.
- [2] M. Elad and A. Feuer, "Super-resolution reconstruction of image sequences," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 21, no. 9, pp. 817–834, Sep. 1999.
- [3] M. Protter, M. Elad, H. Takeda, and P. Milanfar, "Generalizing the non-local-means to super-resolution reconstruction," *IEEE Trans. Image Process.*, vol. 18, no. 1, pp. 36–51, Jan. 2009.
- [4] J. Sun, J. J. Zhu, and M. F. Tappen, "Context-constrained hallucination for image super-resolution," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, Jun. 2010, pp. 231–238.
- [5] Y.-W. Tai, S. Liu, M. S. Brown, and S. Lin, "Super resolution using edge prior and single image detail synthesis," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, Jun. 2010, pp. 2400–2407.
- [6] D. Glasner, S. Bagon, and M. Irani, "Super-resolution from a single image," in *Proc. Int. Conf. Comput. Vision*, 2009, pp. 349–356.
- [7] J. Yang, J. Wright, Y. Ma, and T. Huang, "Image super-resolution as sparse representation of raw image patches," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, Jun. 2008, pp. 1–8.
- [8] X. Li and T. Q. Nguyen, "Markov random field model-based edge-directed image interpolation," *IEEE Trans. Image Process.*, vol. 17, no. 7, pp. 1121–1128, Jul. 2008.
- [9] X. Zhang and X. Wu, "Image interpolation by adaptive 2-D autoregressive modeling and soft-decision estimation," *IEEE Trans. Image Process.*, vol. 17, no. 6, pp. 887–896, Jun. 2008.
- [10] R. Fattal, "Image upsampling via imposed edge statistics," *ACM Trans. Graph.*, vol. 26, no. 3, pp. 95:1–95:8, 2007.
- [11] S. Dai, M. Han, W. Xu, Y. Wu, and Y. Gong, "Soft edge smoothness prior for alpha channel super resolution," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, Jun. 2007, pp. 1–8.

- [12] Y.-W. Tai, W.-S. Tong, and C.-K. Tang, "Perceptually-inspired and edge-directed color image super-resolution," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, Jun. 2006, pp. 1948–1955.
- [13] L. Zhang and X. Wu, "An edge-guided image interpolation algorithm via directional filtering and data fusion," *IEEE Trans. Image Process.*, vol. 15, no. 8, pp. 2226–2238, Aug. 2006.
- [14] H. A. Aly and E. Dubois, "Image up-sampling using total-variation regularization with a new observation model," *IEEE Trans. Image Process.*, vol. 14, no. 10, pp. 1647–1659, Oct. 2005.
- [15] H. Chang, D.-Y. Yeung, and Y. Xiong, "Super-resolution through neighbor embedding," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, Jun.–Jul. 2004, pp. 275–282.
- [16] J. Sun, N.-N. Zheng, H. Tao, and H.-Y. Shum, "Image hallucination with primal sketch priors," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, Jun. 2003, pp. 729–736.
- [17] W. T. Freeman, T. R. Jones, and E. C. Pasztor, "Example-based super-resolution," *IEEE Comput. Graph. Appl.*, vol. 22, no. 2, pp. 56–65, Mar.–Apr. 2002.
- [18] X. Li and M. T. Orchard, "New edge-directed interpolation," *IEEE Trans. Image Process.*, vol. 10, no. 10, pp. 1521–1527, Oct. 2001.
- [19] P. Thvenaz, T. Blu, and M. Unser, "Image interpolation and resampling," in *Handbook of Medical Imaging, Processing and Analysis*. Orlando, FL, USA: Academic Press, Inc., 2000, pp. 393–420.
- [20] W. T. Freeman, E. C. Pasztor, and O. Carmichael, "Learning low-level vision," *Int. J. Comput. Vision*, vol. 40, no. 1, pp. 25–47, Jun. 2000.
- [21] J. Sun, Z. Xu, and H.-Y. Shum, "Image super-resolution using gradient profile prior," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, 2008, pp. 1–8.
- [22] H. S. Hou and H. C. Andrews, "Cubic splines for image interpolation and digital filtering," *IEEE Trans. Acoust. Speech Signal Process.*, vol. 26, no. 6, pp. 508–517, Dec. 1978.
- [23] K. Kim and Y. Kwon, "Example-based learning for singleimage SR and JPEG artifact removal," *Max Planck Insitut fur biologische Kybernetik*, Tübingen, Germany Tech. Rep. 173, 2008.
- [24] G. Freedman and R. Fattal, "Image and video upscaling from local self-examples," *ACM Trans. Graph.*, vol. 28, no. 3, pp. 1–10, 2010.
- [25] M. Irani and S. Peleg, "Motion analysis for image enhancement: Resolution, occlusion and transparency," *J. Visual Commun. Image Representation*, vol. 4, no. 4, pp. 324–335, 1993.
- [26] S. Baker and T. Kanade, "Limits on super-resolution and how to break them," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 9, pp. 1167–1183, Sep. 2002.
- [27] Z. Lin and H.-Y. Shum, "Fundamental limits of reconstruction-based superresolution algorithms under local translation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 1, pp. 83–97, Jan. 2004.
- [28] B. S. Morse and D. Schwartzwald, "Image magnification using level set reconstruction," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, Dec. 2001, pp. 333–340.
- [29] J. Sun, Z. Xu, and H.-Y. Shum, "Gradient profile prior and its applications in image super-resolution and enhancement," *IEEE Trans. Image Process.*, vol. 20, no. 6, pp. 1529–1542, Jun. 2011.
- [30] X. Yan and J. Shen, "Fast gradient-aware upsampling for cartoon video," in *Proc. Int. Conf. Image Anal. Signal Process.*, 2010, pp. 636–639.
- [31] C. Steger, "An unbiased detector of curvilinear structures," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 20, no. 2, pp. 113–125, Feb. 1998.
- [32] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: From error measurement to structural similarity," *IEEE Trans. Image Process.*, vol. 13, no. 4, pp. 600–612, Apr. 2004.
- [33] D. Martin, C. Fowlkes, D. Tal, and J. Malik, "A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics," in *Proc. Int. Conf. Comput. Vision*, Jul. 2001, pp. 416–423.
- [34] Q. Shan, Z. Li, J. Jia, and C.-K. Tang, "Fast image/video upsampling," in *Proc. ACM Trans. Graphic (SIGGRAPH ASIA)*, vol. 27, no. 7. 2008, pp. 153:1–153:7.



Shiming Xiang received the B.S. degree in mathematics from Chongqing Normal University, Chongqing, China, in 1993, the M.S. degree from Chongqing University, Chongqing, in 1996, and the Ph.D. degree from the Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China, in 2004.

From 1996 to 2001, he was a Lecturer with the Huazhong University of Science and Technology, Wuhan, China. He was a Post-Doctorate with the Department of Automation, Tsinghua University, Beijing, until 2006. He is currently an Associate Professor with the National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences. His current research interests include image processing, pattern recognition, and machine learning.



Gaofeng Meng received the B.S. degree in applied mathematics from Northwestern Polytechnical University, Fremont, CA, in 2002, the M.S. degree in applied mathematics from Tianjin University, Tianjin, China, in 2005, and the Ph.D. degree in control science and engineering from Xi'an Jiaotong University, Xi'an, China, in 2009.

He is currently an Assistant Professor with the National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, Beijing, China. His current research interests include image processing, computer vision, and pattern recognition.



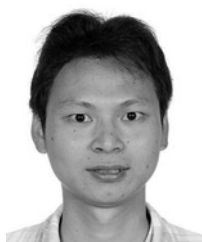
Huaiyu Wu received the B.E. and M.E. degrees from the Beijing University of Aeronautics and Astronautics, Beijing, China, in 2000 and 2003, respectively, and the Ph.D. degree in pattern recognition and intelligent systems from the Chinese Academy of Sciences, Beijing, in 2008.

He is currently an Associate Professor with the National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences. From 2008 to 2011, he was a Post-Doctoral Lecturer with the School of Electronics Engineering and Computer Science, Peking University, Beijing. His current research interests include the fields of 3-D computer vision, visual shape perception and analysis, and interactive computer graphics.



Chunhong Pan received the B.S. degree in automatic control from Tsinghua University, Beijing, China, in 1987, the M.S. degree from the Shanghai Institute of Optics and Fine Mechanics, Chinese Academy of Sciences, Beijing, in 1990, and the Ph.D. degree in pattern recognition and intelligent systems from the Institute of Automation, Chinese Academy of Sciences, in 2000.

He is currently a Professor with the National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences. His current research interests include computer vision, image processing, computer graphics, and remote sensing.



Lingfeng Wang received the B.S. degree in computer science from Wuhan University, Wuhan, China, in 2007. He is currently pursuing the Ph.D. degree with the Department of National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, Beijing, China.

His current research interests include computer vision and image processing.