Super-resolving a Single Blurry Image Through Blind Deblurring Using ADMM

Xiangrong Zeng, Yan Liu, Jun Fan, Qizi Huangpeng, Jing Feng, Jinlun Zhou, Maojun Zhang

College of Information System and Management
National University of Defense Technology
Changsha, China 410073
Email: zengxrong@gmail.com

Abstract—Both single blind image super-resolution (SBISR) and blind image deblurring (BID) are ill-posed inverse problems typically addressed by imposing some form of regularization (prior knowledge) on the unknown blurs and original images (the high resolution image and the sharp image for SBISR and BID, respectively). However, SBISR is more ill-posed than BID due to the introduce of the downsampling operator in the former, thus, the latter is usually easier to be solved than the former. We propose to address the SBISR problem by a BID method via reformulating it into a BID problem by an interpolation operator, and then solving the BID problem using the alternating direction method of multipliers (ADMM). Our approach bridges the gap between SBISR and BID, taking advantages of existing BID methods to handle SBISR. Experiments on synthetic and real blurry images (also on a real sharp image) show that the proposed method is effective, and competitive in terms of speed and restoration quality.

Keywords–Image super-resolution; Blind image deblurring; ADMM.

I. INTRODUCTION

Single image super-resolution (SISR)[1-8] aims at recovering a high-resolution (HR) image $\mathbf{x} \in \mathbb{R}^{N_h}$ from a low-resolution (LR) input image $\mathbf{y} \in \mathbb{R}^{N_l}$ which is defined to be the LR noisy version of the HR image as

$$\mathbf{y} = \mathbf{DBx} + \mathbf{n} \qquad (1)$$

where $\mathbf{D} : \mathbb{R}^{N_h} \to \mathbb{R}^{N_l}(N_l < N_h)$ is the downsampling matrix, $\mathbf{B} : \mathbb{R}^{N_h} \to \mathbb{R}^{N_h}$ is the blurring matrix, and $\mathbf{n} \in \mathbb{R}^{N_l}$ is the additive noise term. The SISR problem is typically severely ill-posed since $\mathbf{DB}$ is rectangular with more columns than rows, and it is more ill-posed than *multi-frame super-resolution* (MFSR) [9][10] and, thus, it can only be solved satisfactorily via regularization by utilizing an image model or prior. If $\mathbf{B}$ is the identity, then (1) reduces to the image interpolation problem [11][12] under noise; if $\mathbf{B}$ is unknown, then (1) evolves to *single blind image super-resolution* (SBISR), which is more complicated than the SISR one and more realistic, and is the focus of this paper. However, most SISR methods assume that $\mathbf{B}$ is known, that is, it is usually predefined, such as Gaussian blur [13], bicubic interpolation [7][8], Gaussian blur followed by bicubic interpolation [14], simple pixel averaging [2], and so on. Only a few works have been dedicated to the SBISR problem. For instance, a parametric Gaussian model with unknown width was assumed for the blur kernel in [13][15][16], and its extension to multiple parametric models was proposed in [17]. A nonparametric model for kernel recovery was presented in [18] via assuming that the kernel has a single peak. All these methods have a restrictive assumption on the blur kernel.

Recently, [19] showed that an accurate blur model is critical to the success of SISR algorithms, and [20] presented that the PSF of the camera is the wrong blur kernel to use in SISR algorithms, and showed how to correct the blur kernel from the LR image. Both [19] and [20] seek accurate blur kernels based on existing SISR algorithms (such as [6][7][8] with complex nature and costly computation), and, thus, their complexities are even more than those of the SISR ones.

In this paper, we address the SBISR problem via a *blind image deblurring* (BID) method, and the rationale behind this idea is that BID is usually easier to be solved than SBISR. The proposed method first reformulates the SBISR problem into a BID one by an interpolation operator, and then handle the BID by alternating minimization, in which, each sub-problem is efficiently solved by the *alternating direction method of multipliers* (ADMM) [21][22]. Thus, the proposed method bridges the gap between SBISR and BID, benefitting from that some BID methods (such as [23-27] and many others omitted here due to space limiation) are arguably faster and easier to understand, than state-of-the-art SISR/SBISR methods, and reaching competitive speed and restoration quality. The paper is organized as follows: Section II introduces the proposed approach, Section III reports experimental results, and Section IV ends the paper with the conclusion.

II. PROPOSED APPROACH

This section introduces how to reformulate a SBISR problem into a BID one and how to solve the resulting BID problem.

A. Problem formulation

Based on the notations in (1), we first introduce the BID problem, which aims at estimating an image $\mathbf{x}$ from a single observed blurry image $\mathbf{z} \in \mathbb{R}^{N_h}$ satisfying a convolutional degradation model

$$\mathbf{z} = \mathbf{Bx} + \mathbf{s} \qquad (2)$$

where $\mathbf{s} \in \mathbb{R}^{N_h}$ is the additive noise. BID is also a severely ill-posed problem since the image $\mathbf{x}$, the blurring matrix $\mathbf{B}$ and the noise $\mathbf{s}$ are all unknown. In order to build the relationship between the problems of BID and SBISR, inserting (2) into (1) yields

$$\mathbf{y} = \mathbf{D}(\mathbf{z} - \mathbf{s}) + \mathbf{n} = \mathbf{Dz} + (\mathbf{n} - \mathbf{Ds}) \qquad (3)$$

which indicates that the SBISR is more ill-posed than the BID, since both aim to recover $\mathbf{x}$ from $\mathbf{y}$ and $\mathbf{z}$, respectively, with unknown $\mathbf{B}$, but the former has fewer known samples than the latter due to the introduce of $\mathbf{D}$ (namely, the length of $\mathbf{y}$ is less than that of $\mathbf{z}$). This inspires that we can solve the SBISR problem in an easier way via reformulating it into a BID problem. The idea is to first interpolate the LR image $\mathbf{y}$ as

$$\mathbf{u} = \mathbf{Uy} = \mathbf{UDBx} + \mathbf{Un} \quad (4)$$

where $\mathbf{U}$ is the interpolation operator (for instance, the *bicubic* or *bilinear* interpolation operators, or other advanced interpolation operators [11][12]), $\mathbf{u} \in \mathbb{R}^{N_h}$ is the interpolation of $\mathbf{y}$. Then we can rewrite (4) as

$$\mathbf{u} = \mathbf{Kx} + \mathbf{e} = \mathbf{Xk} + \mathbf{e} \quad (5)$$

where $\mathbf{K} = \mathbf{UDB} \in \mathbb{R}^{N_h} \times \mathbb{R}^{N_h}$ is the new blurring matrix corresponding to a blur filter $\mathbf{k} \in \mathbb{R}^{N_h}$, and $\mathbf{X} \in \mathbb{R}^{N_h} \times \mathbb{R}^{N_h}$ is the square matrix representing the convolution of image $\mathbf{x}$ with the filter $\mathbf{k}$, and $\mathbf{e} = \mathbf{Un}$ is the interpolation of $\mathbf{n}$.

Thus, instead of super-resolving $\mathbf{x}$ from $\mathbf{y}$ (see (1)), the HR image can be obtained via bind deblurring of $\mathbf{x}$ from $\mathbf{u}$ (see (5)), which becomes the new focus of this paper):

$$(\hat{\mathbf{x}}, \hat{\mathbf{k}}) = \arg\min_{\mathbf{x}, \mathbf{k}} \frac{\lambda}{2} \|\mathbf{Kx} - \mathbf{u}\|_2^2 + \phi_{\text{GTV}}(\mathbf{x}) + \iota_{\mathcal{S}}(\mathbf{k}) \quad (6)$$

where λ is a positive parameter, ϕ_{GTV} a *generalized total variation* (GTV) regularizer given by

$$\phi_{\text{GTV}}(\mathbf{x}) = \|\mathbf{D}_h\mathbf{x}\|_p^p + \|\mathbf{D}_v\mathbf{x}\|_p^p = \sum_i |[\mathbf{D}_h\mathbf{x}]_i|^p + |[\mathbf{D}_v\mathbf{x}]_i|^p$$

where $\mathbf{D}_h$ and $\mathbf{D}_v$ denote the horizontal and vertical derivative partial operator, respectively. Since the distribution of gradients of natural images is more heavy-tailed than Laplace distribution (see [28]), we set $0 \leq p \leq 1$. $\iota_{\mathcal{S}}$ is the indicator function of the set $\mathcal{S}$ which is the probability simplex

$$\mathcal{S} = \{\mathbf{k} : \mathbf{k} \succeq 0, \|\mathbf{k}\|_1 = 1\}. \quad (7)$$

B. Proposed algorithm framework

Alternatively minimizing (6) with respect to $\mathbf{x}$ and $\mathbf{k}$, while increasing the parameter λ, yields the following framework:

Algorithm *Proposed algorithmic framework*
1. **Input**: Observed LR image $\mathbf{y}$, λ and $\alpha > 1$.
2. **Step I**: Interpolate $\mathbf{y}$ via $\mathbf{u} = \mathbf{Uy}$.
3. **Step II**: Blind estimation of blur filter $\mathbf{k}$ from $\mathbf{u}$, by alternative loop over coarse-to-fine levels:
4. ▶ Update the image estimate

$$\hat{\mathbf{x}} \leftarrow \arg\min_{\mathbf{x}} \frac{\lambda}{2}\|\hat{\mathbf{K}}\mathbf{x} - \mathbf{u}\|_2^2 + \phi_{\text{GTV}}(\mathbf{x}) \quad (8)$$

where $\hat{\mathbf{K}}$ is the convolution matrix constructed by $\hat{\mathbf{k}}$ obtained from the blur filter estimation below.
5. ▶ Update the blur filter estimate

$$\hat{\mathbf{k}} \leftarrow \arg\min_{\mathbf{k}} \frac{\lambda}{2}\|\hat{\mathbf{X}}\mathbf{k} - \mathbf{u}\|_2^2 + \iota_{\mathcal{S}}(\mathbf{k}) \quad (9)$$

where $\hat{\mathbf{X}}$ is the convolution matrix constructed by $\hat{\mathbf{x}}$ obtained from the image estimation above.
6. ▶ Increase the parameter λ

$$\lambda \leftarrow \alpha\lambda. \quad (10)$$

7. **Step III**: Non-blind estimation of HR image $\mathbf{x}^*$ from $\mathbf{u}$ through solving (8) with final $\hat{\mathbf{h}}$ (obtained by Step II).
8. **Output**: the HR image $\mathbf{x}^*$ and the blur estimate $\hat{\mathbf{h}}$.

To avoid getting trapped in a local minimum, above algorithmic framework is implemented in a coarse-to-fine fashion as [26][29][30][31]. The sub-problems (8) and (9) can be solved by many existing methods, and next we show how these two sub-problems can be efficiently solved by the ADMM.

C. The ADMM

Before proceeding, we first introduce the ADMM [21][22], which has been as a popular tool to solve imaging inverse problems (see [27][32] and references therein), and is well suited for addressing the general unconstrained minimization problem composed of J sub-functions:

$$\min_{\mathbf{x}} \sum_{j}^{J} g_j(\mathbf{B}^{(j)}\mathbf{x}) \quad (11)$$

where $\mathbf{B}^{(j)}$ are arbitrary matrices and g_j are functions. The ADMM to solve (11) takes the following form (see [32]):

Algorithm *ADMM for solving (11)*
1. Set $k = 0$, $\beta > 0$, $\mathbf{v}_0^{(1)}, \cdots, \mathbf{v}_0^{(J)}, \mathbf{d}_0^{(1)}, \cdots, \mathbf{d}_0^{(J)}$.
2. **repeat**
3. $\quad \mathbf{r}_k = \sum_{j=1}^{J} (\mathbf{B}^{(j)})^T (\mathbf{v}_k^{(j)} + \mathbf{d}_k^{(j)})$
4. $\quad \mathbf{x}_{k+1} = \left[\sum_{j=1}^{J} (\mathbf{B}^{(j)})^T \mathbf{B}^{(j)}\right]^{-1} \mathbf{r}_k$
5. $\quad$ **for** $j = 1, \cdots, J$
6. $\quad\quad \mathbf{v}_{k+1}^{(j)} = \text{Prox}_{g_j/\tau}\left(\mathbf{B}^{(j)}\mathbf{x}_{k+1} - \mathbf{d}_k^{(j)}\right)$
7. $\quad\quad \mathbf{d}_{k+1}^{(j)} = \mathbf{d}_k^{(j)} - (\mathbf{B}^{(j)}\mathbf{x}_{k+1} - \mathbf{v}_{k+1}^{(j)})$
8. $\quad$ **end for**
9. $\quad k \leftarrow k + 1$
10. **until** some stopping criterion is satisfied.

In line 6 of above algorithm, the proximity operator of g_j/τ: $\text{Prox}_{g_j/\tau}$ is defined as

$$\text{Prox}_{g_j/\tau}(\mathbf{v}) = \arg\min_{\mathbf{x}} \left(g_j(\mathbf{x}) + \frac{\tau}{2}\|\mathbf{x} - \mathbf{v}\|^2\right). \quad (12)$$

Next, we tackle the sub-problems (8) and (9) using the ADMM.

D. $\mathbf{x}$ *update using the ADMM*

The sub-problem (8) can be written in the form (11), with

$$g_1(\cdot) = \frac{\lambda}{2}\|\cdot - \mathbf{u}\|_2^2, \ g_2(\cdot) = g_3(\cdot) = \|\cdot\|_p^p, \quad (13)$$
$$\mathbf{B}^{(1)} = \hat{\mathbf{K}}, \ \mathbf{B}^{(2)} = \mathbf{D}_h, \ \mathbf{B}^{(3)} = \mathbf{D}_v \quad (14)$$

then solving (8) using the ADMM yields the following algorithm:

Algorithm *ADMM for solving (8)*
1. **Initialize** $k = 0$, $\tau_1 > 0$, $\mathbf{v}_0^{(1)}, \mathbf{v}_0^{(2)}, \mathbf{v}_0^{(3)}, \mathbf{d}_0^{(1)}, \mathbf{d}_0^{(2)}, \mathbf{d}_0^{(3)}$.
2. **repeat**
3. $\quad \mathbf{z}_k^{(1)} = \mathbf{v}_k^{(1)} + \mathbf{d}_k^{(1)}$
4. $\quad \mathbf{z}_k^{(2)} = \mathbf{v}_k^{(2)} + \mathbf{d}_k^{(2)}$
5. $\quad \mathbf{z}_k^{(3)} = \mathbf{v}_k^{(3)} + \mathbf{d}_k^{(3)}$
6. $\quad \mathbf{r}_k = \hat{\mathbf{K}}^T \mathbf{z}_k^{(1)} + \mathbf{D}_h^T \mathbf{z}_k^{(2)} + \mathbf{D}_v^T \mathbf{z}_k^{(3)}$
7. $\quad \mathbf{x}_{k+1} = \left[\hat{\mathbf{K}}^T\hat{\mathbf{K}} + \mathbf{D}_h^T\mathbf{D}_h + \mathbf{D}_v^T\mathbf{D}_v\right]^{-1} \mathbf{r}_k$

(a) (b) PSNR (dB): 18.008 (c) PSNR(dB): **20.861**
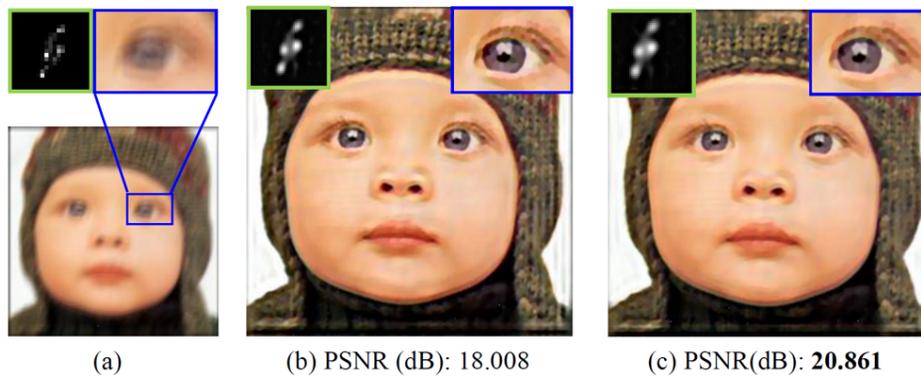
Figure 1. Estimated HR images (size: 512×512), PSFs and PSNRs. (a) are input LR blurry image (size: 256×256, obtained by (1)) and one of the eight PSFs (corresponding to **B**); (b) and (c) are estimated HR images, PSFs (corresponding to **K**) and PSNRs by the proposed method with the *bicubic* and *bilinear* interpolation operators, respectively.

Figure 2. Other seven PSFs and their corresponding estimated PSFs and PSNRs by the proposed method with the *bicubic* and *bilinear* operators, respectively.

8. $\quad \mathbf{v}_{k+1}^{(1)} = \mathrm{Prox}_{g_1/\tau_1}\left(\hat{\mathbf{K}}\mathbf{x}_{k+1} - \mathbf{d}_k^{(1)}\right)$

9. $\quad \mathbf{d}_{k+1}^{(1)} = \mathbf{d}_k^{(1)} - (\hat{\mathbf{K}}\mathbf{x}_{k+1} - \mathbf{v}_{k+1}^{(1)})$

10. $\quad \mathbf{v}_{k+1}^{(2)} = \mathrm{Prox}_{g_2/\tau_1}\left(\mathbf{D}_h\mathbf{x}_{k+1} - \mathbf{d}_k^{(2)}\right)$

11. $\quad \mathbf{d}_{k+1}^{(2)} = \mathbf{d}_k^{(2)} - (\mathbf{D}_h\mathbf{x}_{k+1} - \mathbf{v}_{k+1}^{(2)})$

12. $\quad \mathbf{v}_{k+1}^{(3)} = \mathrm{Prox}_{g_3/\tau_1}\left(\mathbf{D}_v\mathbf{x}_{k+1} - \mathbf{d}_k^{(3)}\right)$

13. $\quad \mathbf{d}_{k+1}^{(3)} = \mathbf{d}_k^{(3)} - (\mathbf{D}_v\mathbf{x}_{k+1} - \mathbf{v}_{k+1}^{(3)})$

14. $\quad k \leftarrow k + 1$

15. **until** some stopping criterion is satisfied.

In above algorithm, line 7 is involved by the inversion of the matrix $\hat{\mathbf{K}}^T\hat{\mathbf{K}} + \mathbf{D}_h^T\mathbf{D}_h + \mathbf{D}_v^T\mathbf{D}_v$, which is block-circulant. Thus, it can be digonalzied by 2D *discrete Fourier transform* (DFT) with $\mathcal{O}(n\log n)$ cost, and the inversion of the resulting diagonal matrix can be computed with $\mathcal{O}(n)$ cost. Line 8 is the proximity operator of g_1/τ_1, which can be obtained in a closed-form:

$$\mathbf{v}_{k+1}^{(1)} = \frac{\lambda\mathbf{u} + \tau_1(\hat{\mathbf{K}}\mathbf{x}_{k+1} - \mathbf{d}_k^{(1)})}{\lambda + \tau_1}; \qquad (15)$$

line 10 and 12 are the proximity operators of the ℓ_p ($0 \le p \le 1$) norm, and they have closed-form solutions for $p \in \{0, \frac{1}{2}, \frac{2}{3}, 1, \frac{4}{3}, \frac{3}{2}, 2\}$ (see [33]). For other general p, no closed-form solution exists, but it can be pre-computed numerically and used in the form of lookup table as that in [28].

E. **k** *update using the ADMM*

In the same vein as above, the sub-problem (9) can be written in the form (11), with

$$g_1(\cdot) = \frac{\lambda}{2}\left\|\cdot - \mathbf{u}\right\|_2^2, \; g_2(\cdot) = \iota_{\mathcal{S}}(\cdot), \qquad (16)$$
$$\mathbf{B}^{(1)} = \hat{\mathbf{X}}, \; \mathbf{B}^{(2)} = \mathbf{I}, \qquad (17)$$

yielding the following algorithm:

Algorithm *ADMM for solving (9)*

1. $\quad$ **Initialize** $k = 0$, $\tau_2 > 0$, $\mathbf{v}_0^{(1)}, \mathbf{v}_0^{(2)}, \mathbf{d}_0^{(1)}, \mathbf{d}_0^{(2)}$.

2. $\quad$ **repeat**

3. $\qquad \mathbf{z}_k^{(1)} = \mathbf{v}_k^{(1)} + \mathbf{d}_k^{(1)}$

4. $\qquad \mathbf{z}_k^{(2)} = \mathbf{v}_k^{(2)} + \mathbf{d}_k^{(2)}$

5. $\qquad \mathbf{r}_k = \hat{\mathbf{X}}^T\mathbf{z}_k^{(1)} + \mathbf{z}_k^{(2)}$

6. $\qquad \mathbf{k}_{k+1} = \left[\hat{\mathbf{X}}^T\hat{\mathbf{X}} + \mathbf{I}\right]^{-1}\mathbf{r}_k$

7. $\qquad \mathbf{v}_{k+1}^{(1)} = \mathrm{Prox}_{g_1/\tau_2}\left(\hat{\mathbf{X}}\mathbf{k}_{k+1} - \mathbf{d}_k^{(1)}\right)$

8. $\qquad \mathbf{d}_{k+1}^{(1)} = \mathbf{d}_k^{(1)} - (\hat{\mathbf{X}}\mathbf{k}_{k+1} - \mathbf{v}_{k+1}^{(1)})$

9. $\qquad \mathbf{v}_{k+1}^{(2)} = \mathrm{Prox}_{g_2/\tau_2}\left(\mathbf{k}_{k+1} - \mathbf{d}_k^{(2)}\right)$

10. $\qquad \mathbf{d}_{k+1}^{(2)} = \mathbf{d}_k^{(2)} - (\mathbf{k}_{k+1} - \mathbf{v}_{k+1}^{(2)})$

11. $\qquad k \leftarrow k + 1$

12. **until** some stopping criterion is satisfied.

In line 6, the matrix $\hat{\mathbf{X}}^T\hat{\mathbf{X}} + \mathbf{I}$ can also be diagonalized by DFT with $\mathcal{O}(n\log n)$ cost. Line 7 can be evaluated in a closed-form
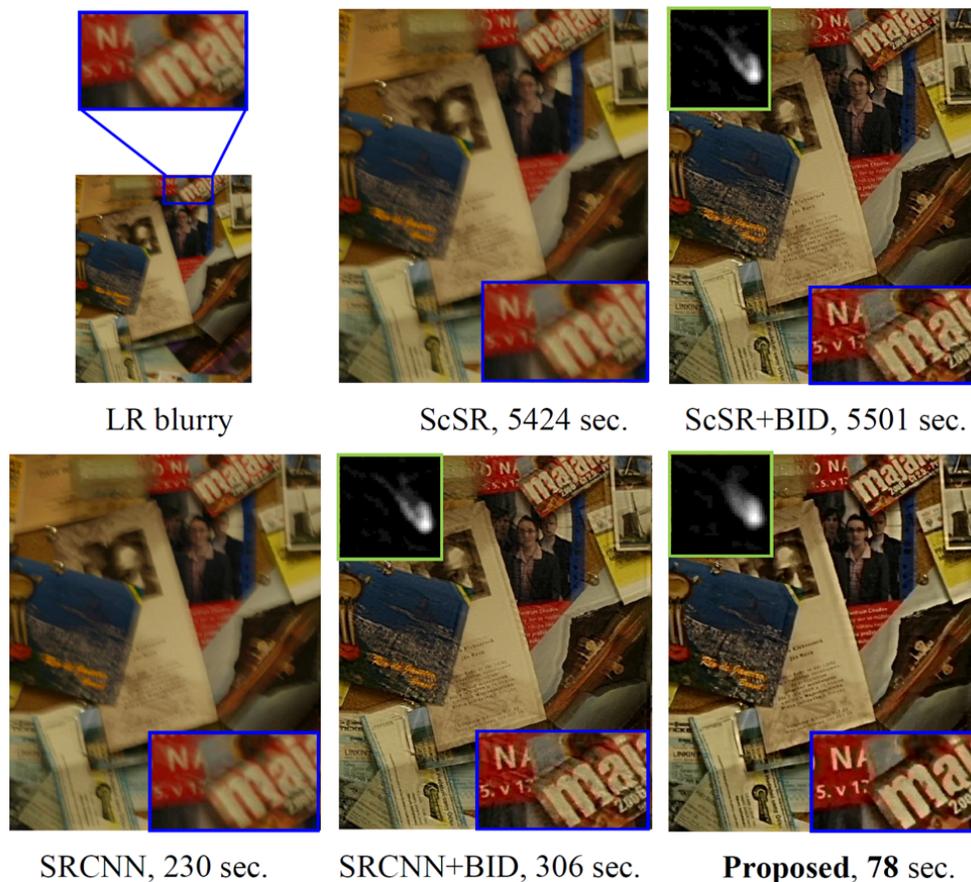
Figure 3. Results on a real LR blurry image, where the LR image size is 900×540 and the HR image size is 1800×1080.

Figure 4. Results on a real LR image, where the LR image size is 324×464 and the HR image size is 648×928.

as (15). Line 9 is the projection onto the probability simplex S (see (7)), which has been already addressed in [34].

III. Experiments

In this section, we report detailed results of the proposed method. All the experiments were performed using MATLAB on a 64-bit Windows 10 personal computer with an Intel Core i7 2.5 GHz processor and 6.0 GB of RAM. The parameters of proposed method are set as $\lambda = 1, \alpha = 1.5, \tau_1 = \tau_2 = 0.15$ and $p = 0.5$, and the setups of involved state-of-the-art methods remain unchanged as their original ones. The BID flow for a color image is: (1) first, convert the image from RGB color space to YCbCr color space, (2) BID of the luminance channel, and (3) convert the image back to the RGB color space. The stopping criterion is chose as $||\hat{\mathbf{f}}_{k+1} - \hat{\mathbf{f}}_k||/||\hat{\mathbf{f}}_{k+1}|| \leq 0.0001$

where $\hat{\mathbf{f}}_k$ is the image estimate or kernel estimate at the k-th iteration. Other parameters are set by following those in [27]. Other details are shown as follows:

A. On synthetic blurry images

In this sub-section, we tested our algorithm on the Baby image (size: 512×512) blurred by eight PSFs of true motion blur provided by [35]. In the algorithm, the operator $\mathbf{U}$ has two options: the *bicubic* and *bilinear* interpolation operators. For saving space, we only show the results on the image blurred by one PSF in Fig. 1, and the results with other seven PSFs are shown in Fig. 2. Notice that PSNR is defined as $20 \log_{10}(255/\sqrt{\text{MSE}})$ where MSE is the mean squared-error between the luminance channel of the original Baby image and the restored HR one. Fig. 1 and 2 verify the rationality of

reformulating SBISR into BID (see (5)).

B. On real images

We also tested our algorithm (only with the *bilinear* interpolation operator due to space limitation) on real images, comparing with state-of-the-art SISR methods: ScSR [8] and SRCNN [36]. Since we currently cannot get access to any SBISR code, we add the proposed BID algorithm as a post-process of the two SISR methods on a real LR blurry image, and the results are shown in Fig. 3. For the sake of fair comparison, we further run the proposed method and the SISR methods on a LR image (not blurry), and the results are shown in Fig. 4. From Fig. 3 and 4, we can see the competitiveness of the proposed method, both in terms of speed and restoration quality, on LR blurry and non-blurry images.

IV. CONCLUSION

We have proposed a new approach for *single blind image super-resolution* (SBISR) via a *blind image deblurring* (BID) method, bridging the gap between SBISR and BID. Experiments on synthetic and real blurry images (also on a real sharp image) show that the effectiveness and competitiveness of the proposed method. Future work will involve exploiting the influence of using advanced interpolation operators on the proposed method.

REFERENCES

[1] H. Chang, D. Y. Yeung, and Y. Xiong, "Super-resolution through neighbor embedding," in IEEE CVPR, 2004, pp. 275–282.

[2] R. Fattal, "Image upsampling via imposed edge statistics," ACM ToG, 2007, vol. 26, pp. 95.

[3] W. Freeman, T. Jones, and E. Pasztor, "Example-based super-resolution," Computer grahics and Applications, vol. 22, no. 2.

[4] S. Mallat and G. Yu, "Super-resolution with sparse mixing estimators," IEEE TSP, vol. 19, 2010, pp. 2889–2900.

[5] J. Yang, J. Wright, T. Huang, and Y. Ma, "Image super-resolution as sparse representation of raw image patches," in IEEE CVPR, 2008.

[6] R. Zeyde, M. Elad, and M. Protter, "On single image scale-up using sparse-representations," Curves and Surfaces, 2012.

[7] D. Glasner, S. Bagon, and M. Irani, "Super-resolution from a single image," in ICCV, 2009, pp. 349–356.

[8] J. Yang, J. Wright, T. S. Huang, and Y. Ma, "Image super-resolution via sparse representation," IEEE TIP, vol. 19, 2010, pp. 2961–2873.

[9] F. Farsiu, D. Robinson, M. Elad, and P. Milanfar, "Fast and roubst multi-frame super-resolution," IEEE TIP, vol. 13, 2003, pp. 1327–1344.

[10] F. Šroubek, G. Cristóbal, and J. Flusser, "A unified approach to superresolution and multichannel blind deconvolution," IEEE TIP, vol. 19, 2010, pp. 2961–2873.

[11] Y. Romano, M. Protter, and M. Elad, "Sparse representation based image interpolation with nonlocal autoregressive modeling," IEEE TIP, vol. 22, 2013, pp. 1382–1394.

[12] X. Zhang and X. Wu, "Image interpolation by adaptive 2-d autoregressive modeling and soft-decision estimation," IEEE TIP, vol. 17, 2008, pp. 887–896.

[13] I. Bégin and F. Ferrie, "Blind super-resolution using a learning-based approach," in ICPR, 2004.

[14] W. Freeman and C. Liu, "Markow random fields for super-resolution and texture synthesis," in Advances in Markov Random Fields for Vision and Image Processing, 2011.

[15] J. Qiao, J. Liu, and C. Zhao, "A novel svm-based blind super-resolution algorithm," in International Joint Conference on Neural Networks, 2006.

[16] Q. Wang, X. Tang, and H. Shum, "Patch based blind image super-resolution," in ICCV, 2005.

[17] Y. He, K. H Yap, L. Chen, and L. P. Chau, "A soft map framework for blind super-resolution reconstruction," Image and Vision Computing, vol. 27, 2009, pp. 364–373.

[18] N. Joshi, R. Szeliski, and D. J. Kriegman, "Psf estimation using sharp edge prediction," in IEEE CVPR, 2008.

[19] N. Efrat, D. Glasner, A. Apartsin, B. Nadler, and A. Levin, "Accurate blur models vs. image priors in single image super-resolution," in ICCV, 2013.

[20] T. Michaeli and M. Irani, "Nonparametric blind super-resolution," in ICCV, 2013.

[21] D. Gabay and B. Mercier, "A dual algorithm for the solution of nonlinear variational problems via nite element approximations," Computers and Mathematics with Applications, vol. 2, 1976, pp. 17–40.

[22] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical learning via the alternating direction method of multipliers," Foundations and Trends in Machine Learning, vol. 3, 2011, pp. 1–122.

[23] R. Fergus, B. Singh, A. Hertzmann, S. Roweis and W. Freeman, "Removing camera shake from a single photograph," in ACM Transactons on Graph, 2006, pp. 787–794.

[24] P. Campisi and K. Egiazarian, "Blind image deconvolution: theory and applications," in CRC press., 2007.

[25] L. Xu and J. Jia, "Two-phase kernel estimation for robust motion deblurring," in ECCV, 2010, pp. 157–170.

[26] J. Kotera, F. Sroubek, and P. Milanfar, "Blind deconvolution using alternating maximum a posteriori estimation with heavy-tailed priors," in Computer Analysis of Images and Patterns, Lecture Notes in Computer Science, vol. 8048, 2013, pp. 59–66.

[27] M. S. C. Almeida, and M. A. T. Figueiredo, "Blind image deblurring with unknown boundaries using the alternating direction method of multipliers," ICIP, 2013, pp. 586–590.

[28] D. Krishnan, and R. Fergus, "Fast image deconvolution using hyper-laplacian priors", 2009.

[29] S. Cho and S. Lee, "Fast motion deblurring," ACM ToG, vol. 28, 2009.

[30] C. Wang, L. Sun, Z. Chen, J. Zhang, and S. Yang, "Multi-scale blind motion deblurring using local minimum," Inverse Problems, vol. 26, 2010, pp. 015003.

[31] D. Krishnan, T. Tay, and R. Fergus, "Blind deconvolution using a normalized sparsity measure," in IEEE CVPR, 2011, pp. 233–240.

[32] M. V. Afonso, J. M. Bioucas-Dias, and M. A. T. Figueiredo, "An augmented lagrangian approach to the constrained optimization formulation of imaging inverse problems," IEEE TIP, vol. 20, 2011, pp. 681–695.

[33] P. L. Combettes and V. R. Wajs, "Signal recovery by proximal forward-backward splitting," Multiscale Modeling & Simulation, vol. 4, 2005, pp. 1168–1200.

[34] J. Duchi, S. Shalev-Shwartz, Y. Singer, and T. Chandra, "Efficient projections onto the l1-ball for learning in high dimensions," in Proc. 25th international conference on Machine learning, 2008, pp. 272–279.

[35] A. Levin, Y. Weiss, F. Durand, and W. T. Freeman, "Understanding and evaluating blind deconvolution algorithms," in IEEE CVPR, 2009, pp. 1964–1971.

[36] C. Dong, C. C. Loy, K. He, and X. Tang, "Learning a deep convolution network for image super-resolution," in ECCV, 2014.