Additional Experimental Results

1. Experimental results on image deconvolution

In this section, we provide more experimental results to evaluate the GISA method for image deconvolution. Followed the experiment setting in [21], we use 8 clean images (im1 ~ im8 as shown in the main paper) and five real world camera shake kernels (kernel1 ~ kernel5 as shown in the main paper) to generate the blurry images. Gaussian white noise with variance of 0.01 is further added to the blurry images. Then we compare GISA with LUT [21], IRLS [24], IRL1 [8], and ITM-l_p [28] in terms of PSNR, running time, and energy function value.

Figure S-1. Blurry images of im1: (a) original image, and the blurry images by (b) kernel1, (c) kernel2, (d) kernel3, (e) kernel4, and (f) kernel5.

Fig. S-1 shows the five blurry images of im1 obtained by using kernel1 ~ kernel5, respectively. Fig. S-2 ~ Fig. S-6 show the deconvolution results by GISA, LUT [21], IRLS [24], IRL1 [8], and ITM-l_p [28], respectively, while Table S-1 ~ Table S-5 list the quantitative PSNR, running time, and energy function values of different methods, respectively. Compared with IRLS, IRL1 and ITM-l_p, GISA can achieve higher PSNR values and is computationally more efficient, and the $F(x)$ value of it is lower. We argue that GISA converges to a better minimum, and this might be the reason of its relatively higher PSNR values. GISA and LUT obtain similar PSNR and $F(x)$ values, which indicate that these two methods lead to similar solutions. Compared with LUT, GISA is more efficient, and does not require the generation and storage of the look-up table.

Figure S-2. Deconvolution results on images (im1) blurred by kernel1 by using: (a) GISA, (b) LUT, (c) IRLS, (d) IRL1, and (e) ITM-l_p.

Table S-1: The PSNR, running time, and function values (im1, kernel1).

Method	IRLS	IRL1	ITM-l_p	LUT	GISA
PSNR	31.15	31.86	31.76	**31.95**	**31.96**
Time (*s*)	117.60	179.21	0.58	0.54	**0.37**
Energy $F(\mathbf{x})$	22.18	20.71	20.74	**20.59**	**20.60**

(a) (b) (c)

(d) (e)

Figure S-3. Deconvolution results on images (im1) blurred by kernel2 by using: (a) GISA, (b) LUT, (c) IRLS, (d) IRL1, and (e) ITM-l_p.

Table S-2: The PSNR, running time, and function values (im1, kernel2).

Method	IRLS	IRL1	ITM-l_p	LUT	GISA
PSNR	31.98	**32.53**	32.33	**32.53**	**32.55**
Time (s)	122.50	181.36	0.57	0.56	**0.38**
Energy $F(\mathbf{x})$	21.84	20.70	20.66	**20.59**	**20.63**

(a) (b) (c)

(d) (e)

Figure S-4. Deconvolution results on images (im1) blurred by kernel3 by using: (a) GISA, (b) LUT, (c) IRLS, (d) IRL1, and (e) ITM-l_p.

Table S-3: The PSNR, running time, and function values (im1, kernel3).

Method	IRLS	IRL1	ITM-l_p	LUT	GISA
PSNR	32.48	33.16	32.93	**33.27**	**33.26**
Time (s)	121.07	181.58	0.56	0.55	**0.37**
Energy $F(\mathbf{x})$	22.02	20.84	20.82	**20.71**	**20.72**

(a) (b) (c)

(d) (e)

Figure S5. Deconvolution results on images (im1) blurred by kernel4 by using: (a) GISA, (b) LUT, (c) IRLS, (d) IRL1, and (e) ITM-l_p.

Table S-4: The PSNR, running time, and function values (im1, kernel4).

Method	IRLS	IRL1	ITM-l_p	LUT	GISA
PSNR	33.18	33.88	33.64	**33.93**	**33.92**
Time (s)	119.97	183.17	0.56	0.55	**0.38**
Energy $F(\mathbf{x})$	22.83	21.58	21.58	**21.42**	**21.46**

Figure S-6. Deconvolution results on images (im1) blurred by kernel5 by using: (a) GISA, (b) LUT, (c) IRLS, (d) IRL1, and (e) ITM-l_p.

Table S-5: The PSNR, running time, and function values (im1, kernel5).

Method	IRLS	IRL1	ITM-l_p	LUT	GISA
PSNR	28.94	28.95	29.27	**29.49**	**29.50**
Time (s)	140.32	188.56	0.60	0.88	**0.37**
Energy $F(\mathbf{x})$	21.84	21.79	20.51	**20.29**	**20.32**

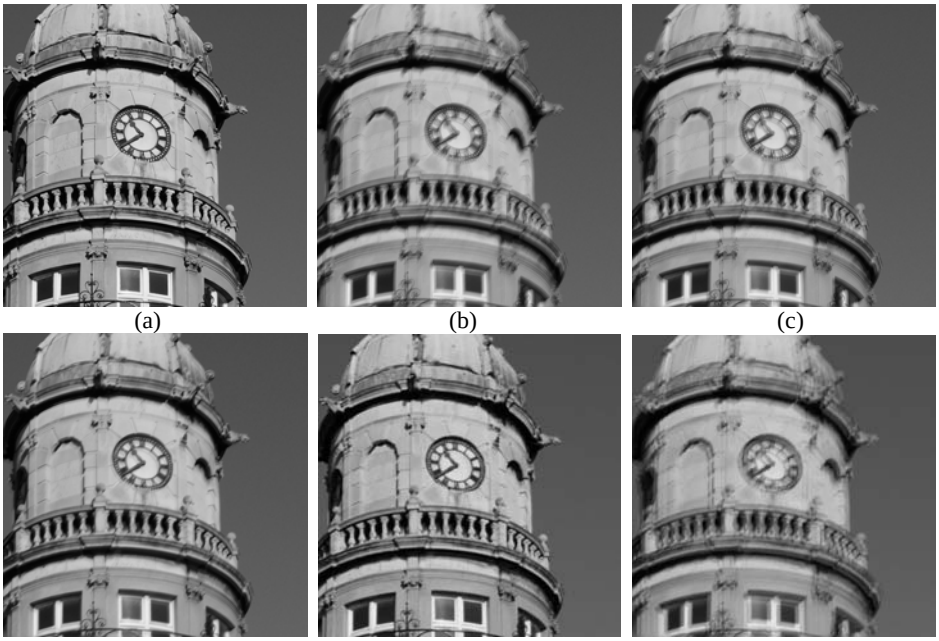


Figure S-7 Blurry images of im8: (a) original image, and the blurry images blurred with (b) kernel1, (c) kernel2, (d) kernel3, (e) kernel4, and (f) kernel5

We also provide here the deconvolution results of GISA, LUT [21], IRLS [24], IRL1 [8], and ITM-l_p [28] on im8. Since the size of im8 is huge (4032 × 6048), for illustrating the restoration results, we crop a 1024 × 1024 subimage to test the algorithms. Fig. S-7 shows the five blurry images of im8 by using kernel1 ~ kernel5, respectively.

Fig. S-8 ~ Fig. S-12 show the deconvolution results of GISA, LUT [21], IRLS [24], IRL1 [8], and ITM-l_p [28], respectively, while Table S-6 ~ Table S-10 list the quantitative PSNR, running time, and energy function values of different methods, respectively. From the figures and tables, we can obtain similar conclusions on GISA.

(a) (b) (c) (d) (e)

Figure S-8. Deconvolution results on images (im8) blurred by kernel1 by using: (a) GISA, (b) LUT, (c) IRLS, (d) IRL1, and (e) ITM-l_p.

Table S-6: The PSNR, running time, and function values (im8, kernel1).

Method	IRLS	IRL1	ITM-l_p	LUT	GISA
PSNR	30.39	31.16	31.09	**31.28**	**31.29**
Time (s)	626.15	958.32	3.05	2.39	**1.92**
Energy $F(\mathbf{x})$	99.16	92.36	91.96	**91.41**	**91.36**

Figure S-9. Deconvolution results on images (im8) blurred by kernel2 by using: (a) GISA, (b) LUT, (c) IRLS, (d) IRL1, and (e) ITM-l_p.

Table S-7: The PSNR, running time, and function values (im8, kernel2).

Method	IRLS	IRL1	ITM-l_p	LUT	GISA
PSNR	31.42	32.05	31.99	**32.12**	**32.13**
Time (*s*)	677.04	986.53	2.93	2.42	**1.89**
Energy $F(\mathbf{x})$	97.66	91.65	91.50	**90.92**	**90.88**

Figure S-10. Deconvolution results on images (im8) blurred by kernel3 by using: (a) GISA, (b) LUT, (c) IRLS, (d) IRL1, and (e) ITM-l_p.

Table S-8: The PSNR, running time, and function values (im8, kernel3).

Method	IRLS	IRL1	ITM-l_p	LUT	GISA
PSNR	32.04	32.83	32.62	**32.94**	**32.95**
Time (s)	661.20	974.38	3.00	2.54	**1.89**
Energy $F(\mathbf{x})$	98.56	92.63	92.61	**91.91**	**91.72**

Figure S-11. Deconvolution results on images (im8) blurred by kernel4 by using: (a) GISA, (b) LUT, (c) IRLS, (d) IRL1, and (e) ITM-l_p.

Table S-9: The PSNR, running time, and function values (im8, kernel4).

Method	IRLS	IRL1	ITM-l_p	LUT	GISA
PSNR	32.81	**33.50**	33.23	**33.52**	**33.53**
Time (*s*)	651.48	994.66	2.94	2.40	**1.92**
Energy $F(\mathbf{x})$	99.93	94.03	93.73	**93.19**	**93.20**

(a) (b) (c)

(d) (e)

Figure S-12. Deconvolution results on images (im8) blurred by kernel5 by using: (a) GISA, (b) LUT, (c) IRLS, (d) IRL1, and (e) ITM-l_p.

Table S-10: The PSNR, running time, and function values (im8, kernel5).

Method	IRLS	IRL1	ITM-l_p	LUT	GISA
PSNR	28.96	29.67	29.67	**29.87**	**29.84**
Time (*s*)	663.84	974.10	2.96	2.40	**2.06**
Energy $F(\mathbf{x})$	98.65	92.06	91.89	**90.92**	**90.84**

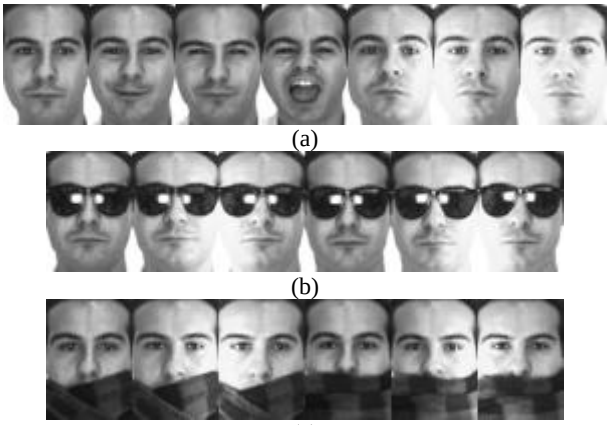
At last, we list the average PSNR, running time, and energy function values over all the eight images and all the five kernels in Table S-11.

Table S-11: The average PSNR, running time, and energy function values over all the 8 images and all the 5 kernels.

Method	IRLS	IRL1	ITM-l_p	LUT	GISA
PSNR	29.89	30.38	30.47	30.65	**30.66**
Time (*s*)	218.38	308.66	0.97	0.86	**0.61**
Energy $F(\mathbf{x})$	34.54	32.96	32.82	32.50	**32.48**

2. Experimental results on robust face recognition with real disguise

In this section, we use the real face images with disguise in the AR face database [S1] to evaluate $\text{SRC}_{p,q}$. Followed by the experimental setting in [32], the subset used in our experiment contains 700 uncorrupted images of 100 subjects (50 males and 50 females), 600 images with sunglasses, and 600 images with scarf. The image size was cropped to 60×43. The 700 uncorrupted images for are used for training, and the 600 images with sunglasses and the 600 images with scarf are used for test. Fig. S-13 shows the seven training images, the six images with sunglasses, and the six images with scarf of one subject.



(a) (b) (c)

Figure S-13. The face images of one subject from the AR database. (a) The seven training images; (b) the six images with glasses; and (c) the six images with scarf.

By setting $p = q$ and varying p, in Fig. S-14 we show the recognition rate of $SRC_{p,q}$ versus different p values. One can see that, when $p = q = 0.5$, $SRC_{p,q}$ can achieve the highest recognition rate for the recognition of images with both sunglasses and scarf. For recognition of images with sunglasses, the recognition rate of $SRC_{p,q}$ with $p = q = 0.5$ is 71.50%, which is 2% higher than that of $SRC_{p,q}$ with $p = q = 1$ (69.50%), i.e., original SRC [32]. For recognition of images with scarf, the recognition rate of $SRC_{p,q}$ with $p = q = 0.5$ is 70.83%, which is also much higher than that of $SRC_{p,q}$ with $p = q = 1$ (69.17%).

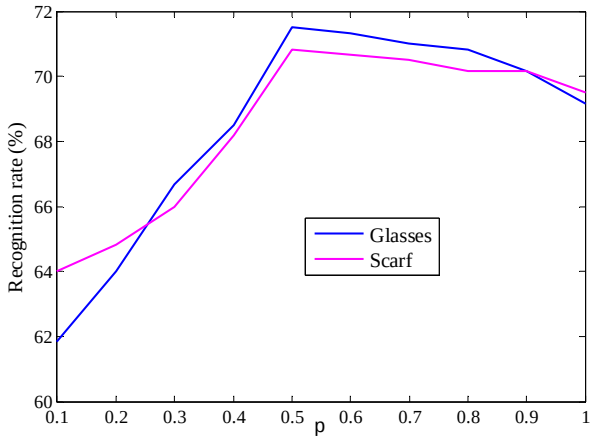


Figure S-14. The recognition rate of $SRC_{p,q}$ for the recognition of images with glasses or scarf by varying the value of p (q=p).

[S1] A. Martinez and R. Benavente. The AR face database. Technical Report 24, CVC, 1998.