A Comparative Study for Single Image Blind Deblurring Supplementary Material

Wei-Sheng Lai[1] Jia-Bin Huang[2] Zhe Hu[1] Narendra Ahuja[2] Ming-Hsuan Yang[1]
[1]University of California, Merced [2]University of Illinois, Urbana-Champaign
http://vllab.ucmerced.edu/~wlai24/cvpr16_deblur_study

1. Overview

In this supplementary document, we present three main additional results to complement the paper. First, we present test images in our real and synthetic datasets, and screenshots of the user study survey webpage. Second, we show complete results of the user study analysis, which include the B-T scores, grouping of algorithms, the percentage of obtained votes and the rationale analysis. Finally, we present qualitative comparisons of state-of-the-art deblurring algorithms on 10 selected blurred images from our real dataset.

2. Datasets

Real Dataset. We show the 100 blurred images of our real dataset in Figure 1. These images cover a wide variety of scenes and challenges for deblurring, allowing us to evaluate the deblurring algorithms in the wild.

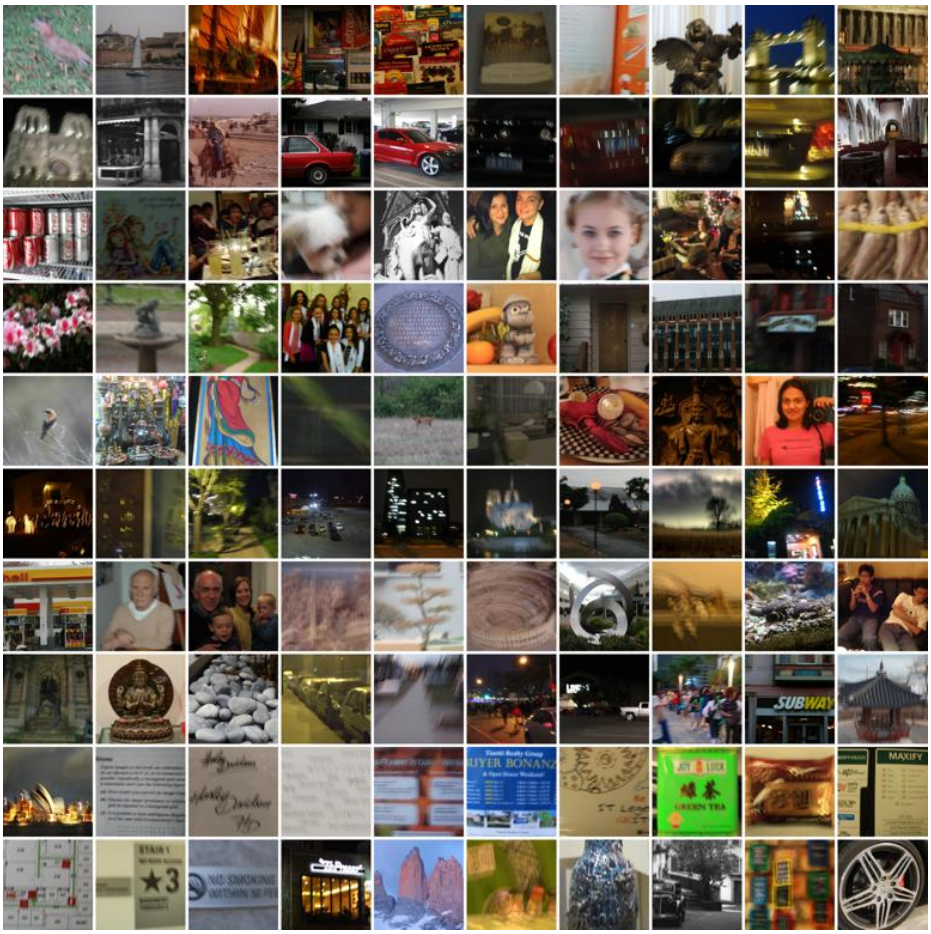


Figure 1: All blurred images in our real dataset.

Synthetic Dataset. Our synthetic dataset consists of 25 latent images (Figure 2), 4 uniform blur kernels (Figure 3), and 4 camera trajectories, i.e., non-uniform blur kernels (Figure 4).

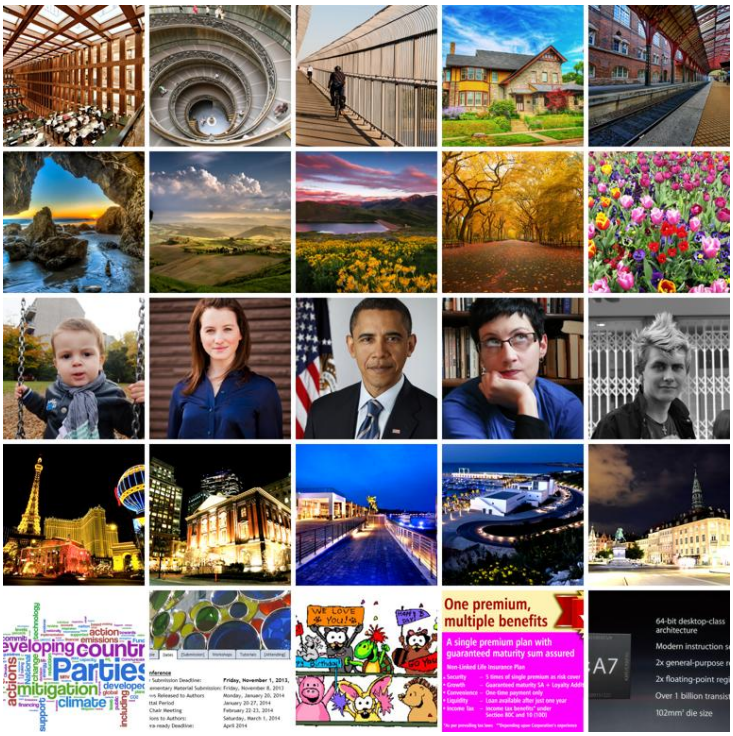
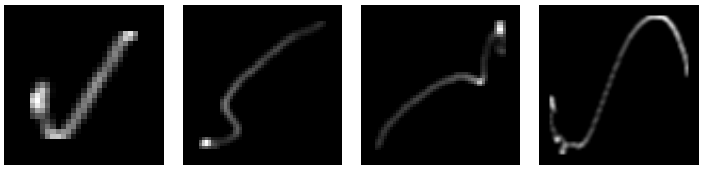


Figure 2: All latent images in our synthetic dataset.



(a) 51×51 (b) 75×75 (c) 75×75 (d) 101×101

Figure 3: Uniform blur kernels generated from [9]. The captions indicate the corresponding support size of the blur kernels.

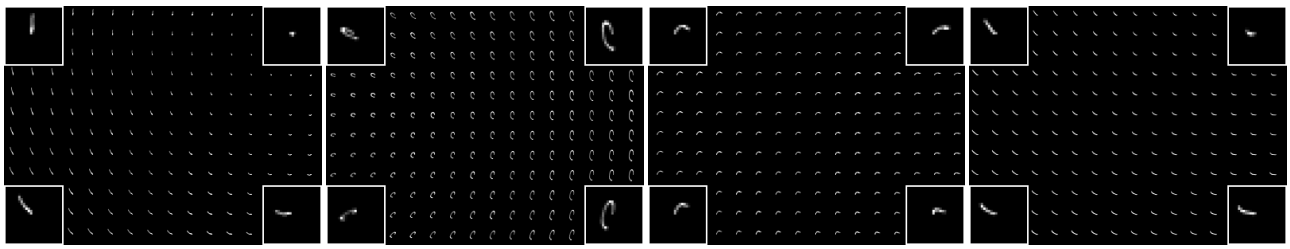


Figure 4: Visualization of the local blur kernels generated from our recorded camera trajectories. The support size of the blur kernel is 25×25.

3. Evaluated Algorithms

In Table 1, we list the details of the evaluated algorithms, including the blur model assumptions, image priors, blur kernel priors, and the execution time under different blur kernel sizes. The execution time is measured with an image of size 800×600 on a machine with 3.6 GHz CPU and 16G RAM.

Table 1: List of the evaluated algorithms. For the blur model column, U: uniform deblurring, NU: non-uniform deblurring. For the code column, E: executable binary code, M: MATLAB M code, P: MATLAB P code.

Method	Blur Model	Image Priors	Blur Kernel Priors	Code	Execution time (sec.) and kernel width			
					25	51	75	101
Fergus-06 [3]	U	mixture-of-Gaussian	mixture-of-exponential	M	218.4	173.5	163.7	146.9
Cho-09 [2]	U	edge selection + L_2 gradient	L_2	E	5.0	7.4	11.3	7.4
Xu-10 [12]	U	edge selection + L_2 gradient	$L_2 + L_1$	E	18.2	22.8	26.4	38.2
Krishnan-11 [4]	U	normalized sparsity	L_1	M	145.2	214.2	312.4	466.1
Levin-11 [5]	U	sparse mixture-of-Gaussian	none	M	385.8	1490.7	9738.0	34014
Whyte-12 [11]	NU	mixture-of-Gaussian	mixture-of-exponential	M	516.2	306.0	445.3	487.0
Sun-13 [10]	U	external patch	L_2	M	1317.1	1335.3	1635.7	1572.1
Xu-13 [13]	U	L_0 gradient	L_2	P	3.3	3.5	4.4	3.6
Zhang-13 [14]	U	weighted L_2 gradient	weighted L_2	P	631.2	2215.7	15258	47053
Zhong-13 [15]	U	non-local mean filter	L_2	P	18.2	20.1	20.4	22.3
Michaeli-14 [6]	U	internal patch	L_2	P	12.9	130.9	2115.9	14045
Pan-14 [7]	U	L_0 intensity + L_0 gradient	L_2	M	170.3	216.0	384.0	319.8
Perrone-14 [8]	U	L_2 gradient	none	M	5745.1	8058.8	10177	10693

4. Human Subject Study

We show screenshots of our user study website in Figure 5 and 6. We first give a brief introduction to the image deblurring task and list two common artifacts (e.g. ringing artifacts, noisy images) before subjects start the survey. In our main survey page, we ask subjects to choose a preferred image from a pair of deblurred results on the screen. Our webpage allows subjects to easily flip between two images by hovering the mouse over the two thumbnails on the top, and show the magnified images in the middle of the page (see Figure 6).

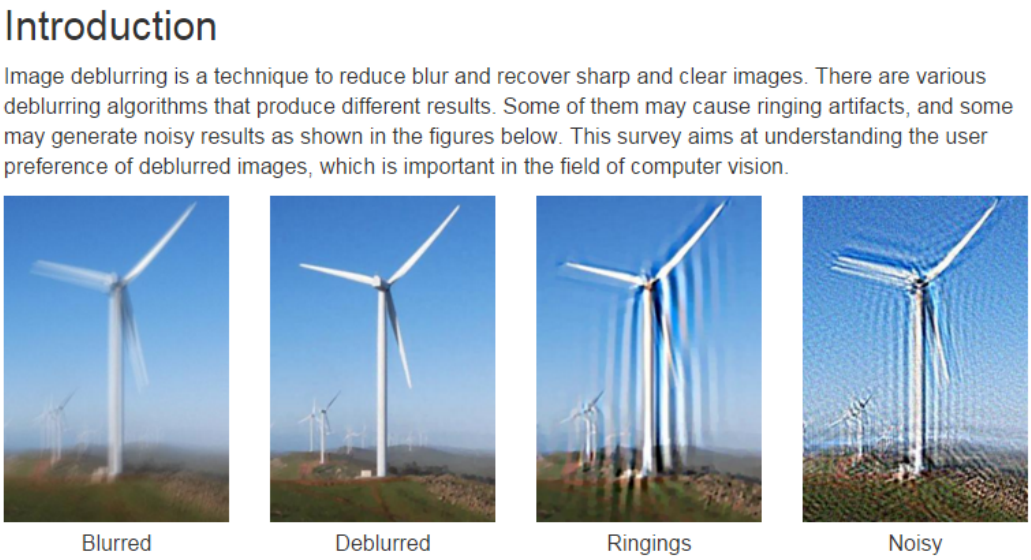


Figure 5: A screenshot of the introduction page in our user study website.

Figure 6: A screenshot of the survey page in our user study website.

5. Ranking

5.1. Ranking from the B-T Scores

Figure 7, 8, and 9 illustrate the complete rankings from the B-T scores [1] on our real, synthetic uniform and synthetic non-uniform datasets, respectively.

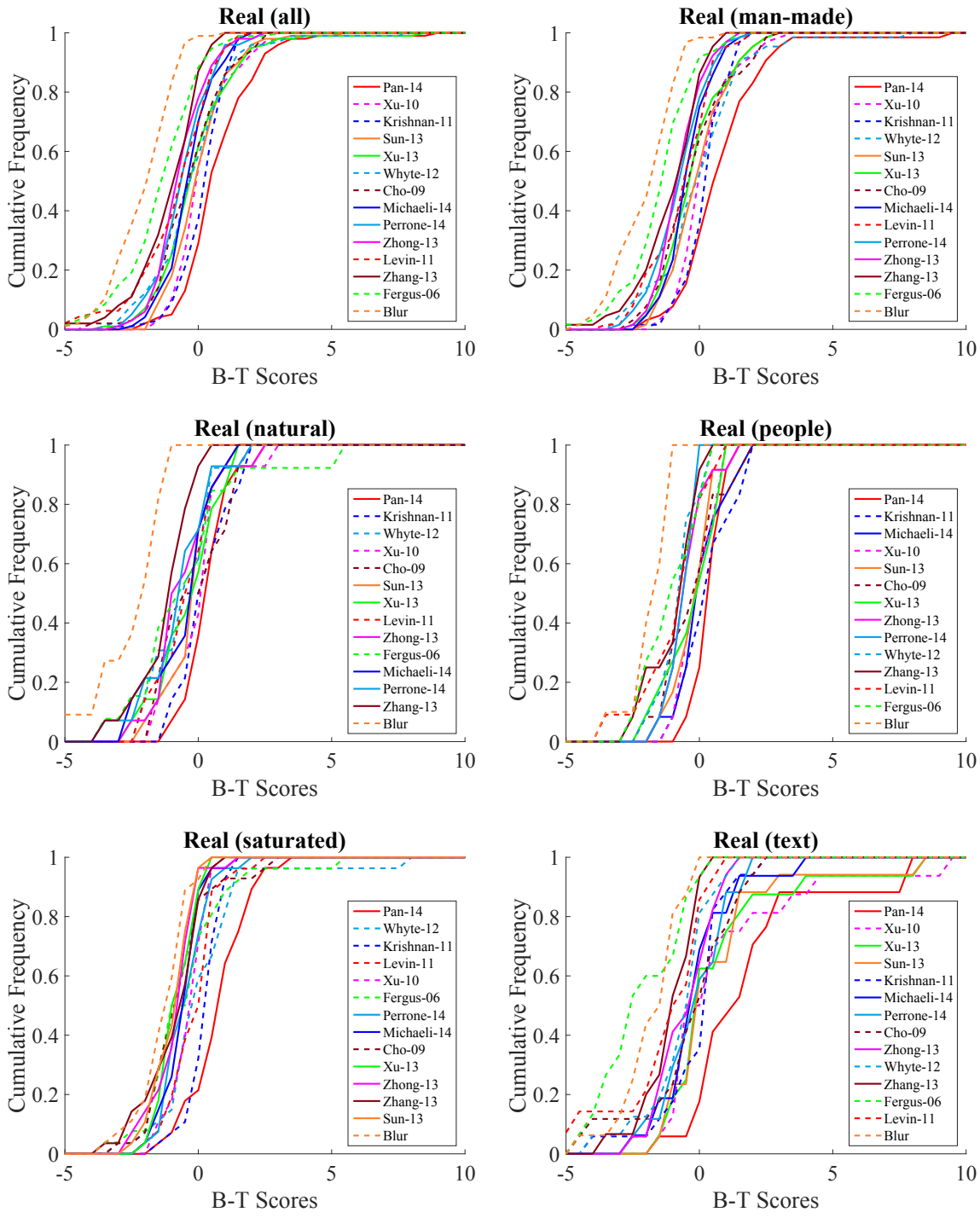


Figure 7: Cumulative frequency and ranking of the B-T scores on our real dataset.

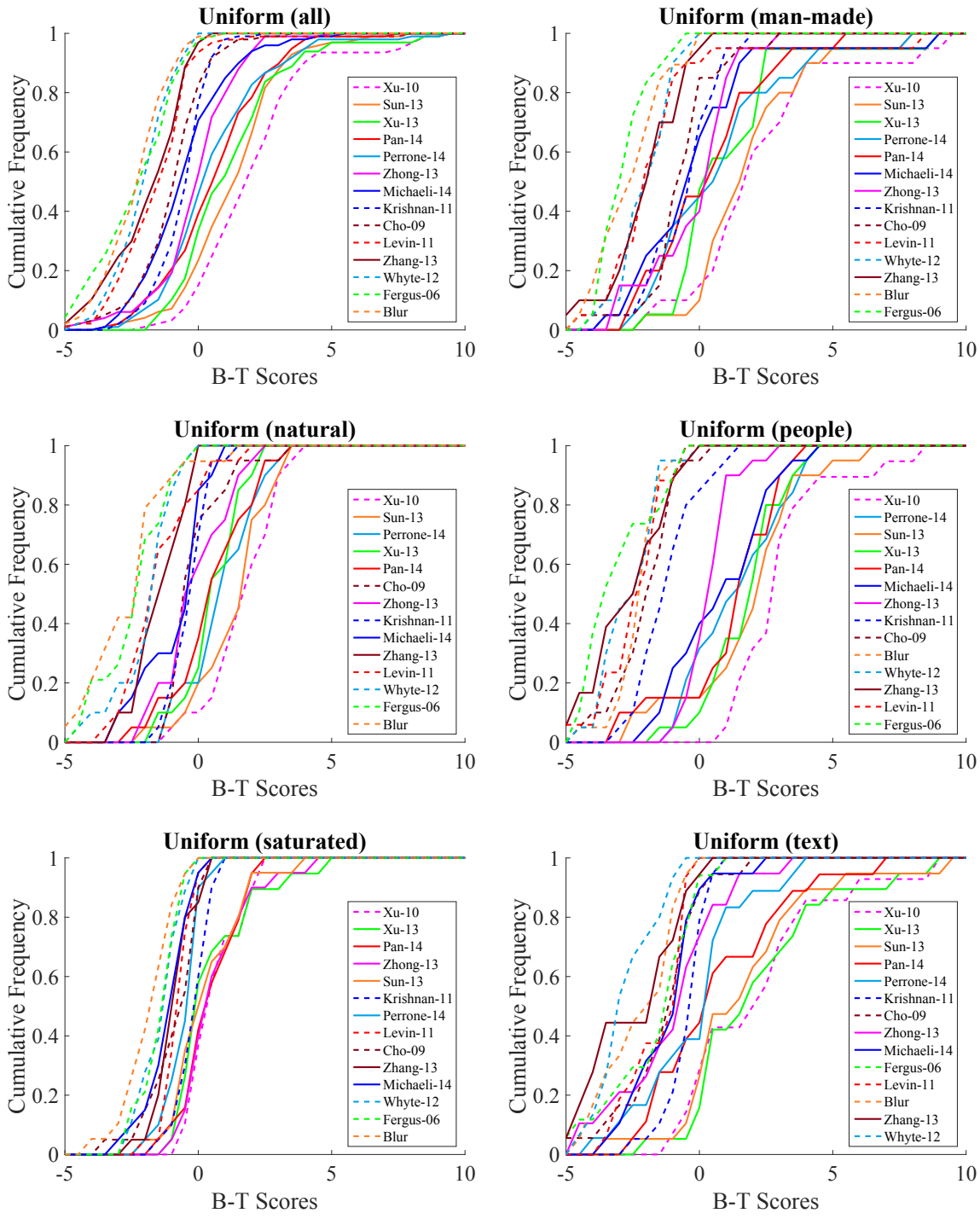


Figure 8: Cumulative frequency and ranking of the B-T scores on our synthetic uniform dataset.

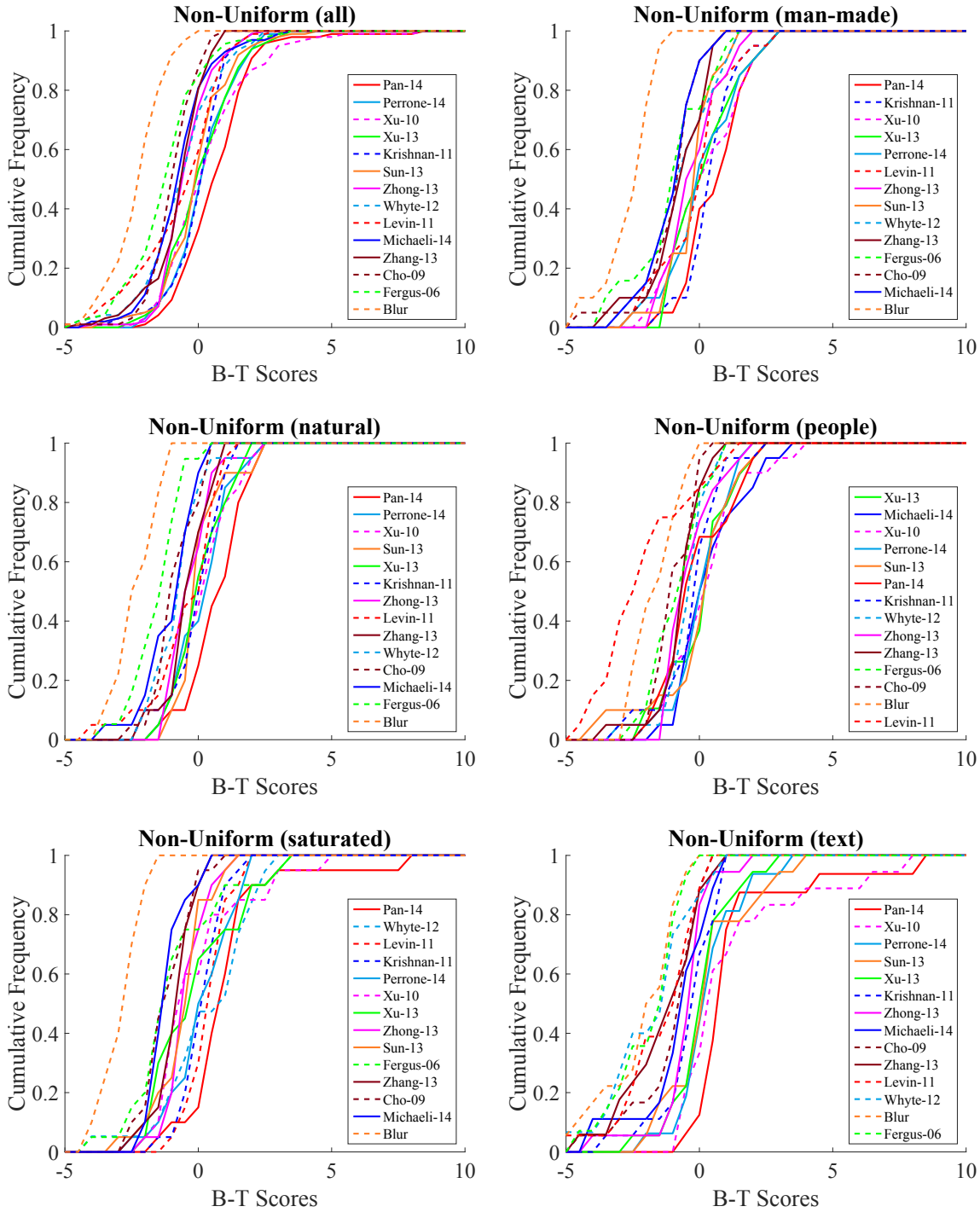
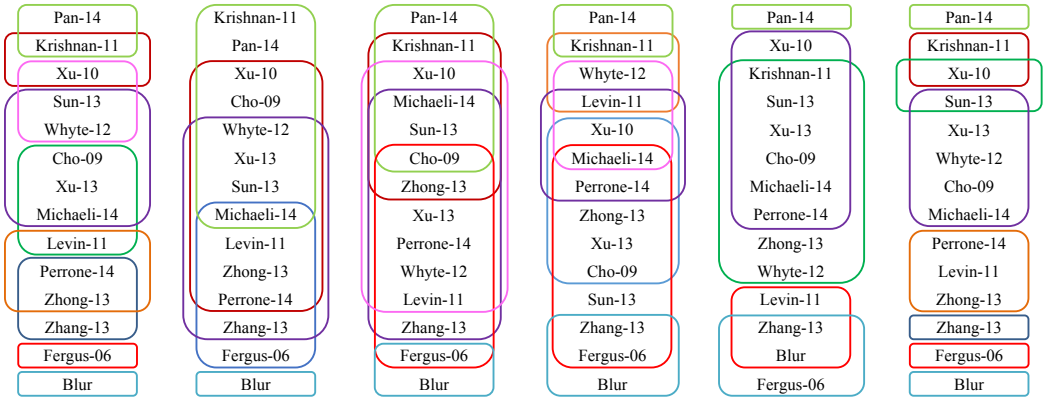


Figure 9: Cumulative frequency and ranking of the B-T scores on our synthetic non-uniform dataset.

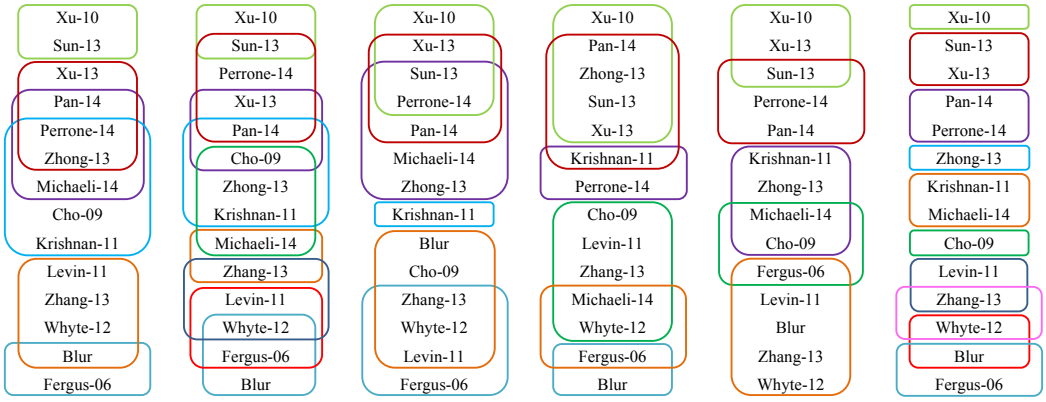
5.2. Ranking from Obtained Votes and Method Grouping

Figure 10, 11, and 12 show the complete rankings from the obtained votes and method grouping from the significance test on our real, synthetic uniform, and synthetic non-uniform datasets, respectively.



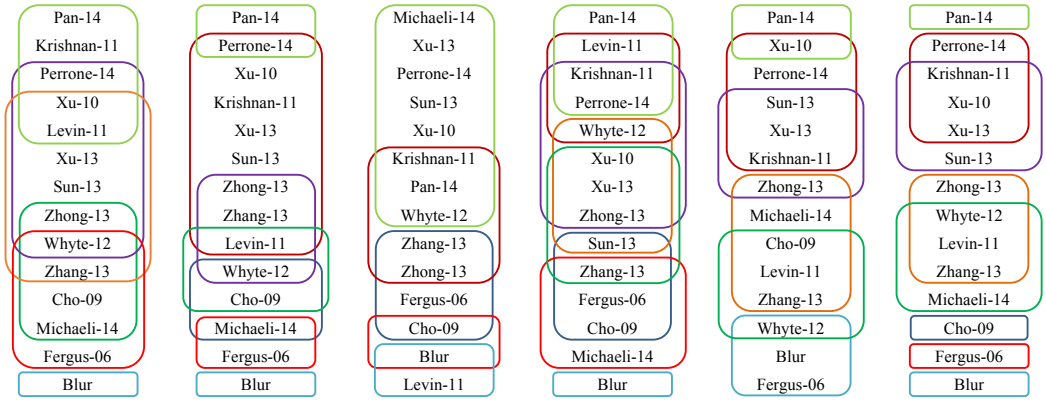
(a) man-made (b) natural (c) people/face (d) saturated (e) text (f) all

Figure 10: Grouping of algorithms by significance test on our real dataset.



(a) man-made (b) natural (c) people/face (d) saturated (e) text (f) all

Figure 11: Grouping of algorithms by significance test on our synthetic uniform dataset.



(a) man-made (b) natural (c) people/face (d) saturated (e) text (f) all

Figure 12: Grouping of algorithms by significance test on our synthetic non-uniform dataset.

Figure 13, 14, and 15 show the percentage of the obtained votes for each evaluated method. Each column represents the performance of evaluated methods under a specific attribute. Each row represents the performance of a specific method under different attributes.

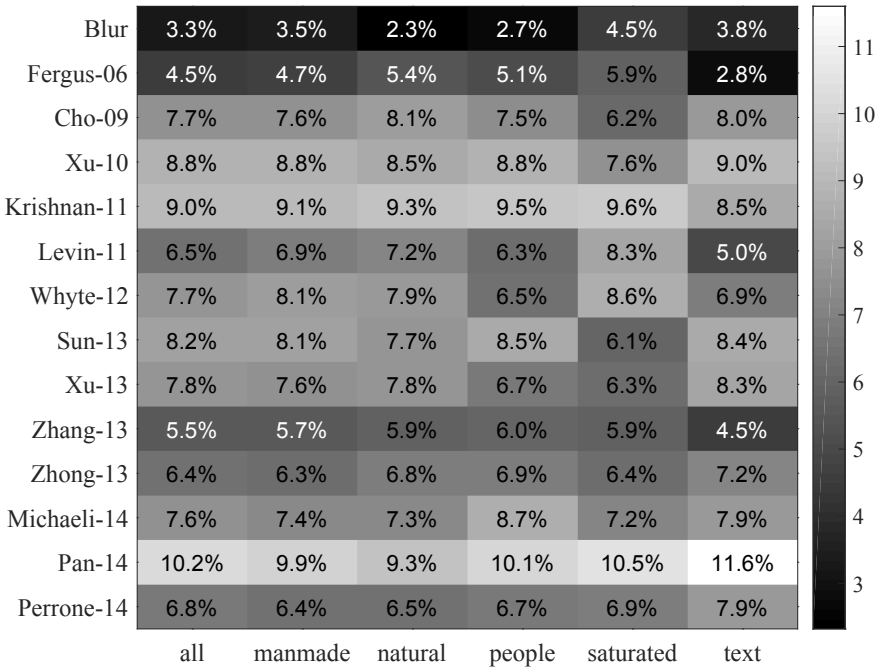


Figure 13: Percentage of the obtained votes with different attributes on our real dataset.

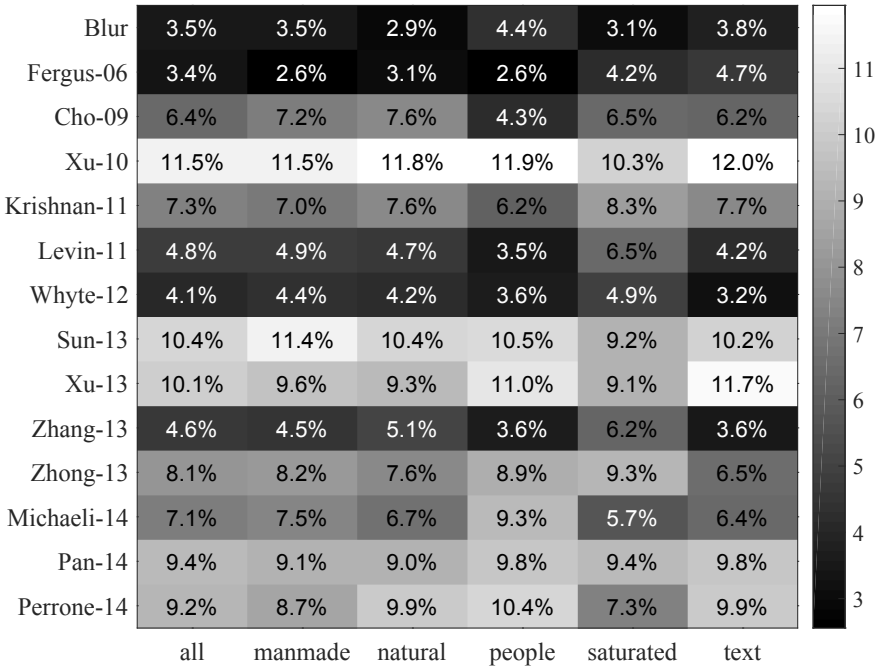


Figure 14: Percentage of the obtained votes with different attributes on our synthetic uniform dataset.

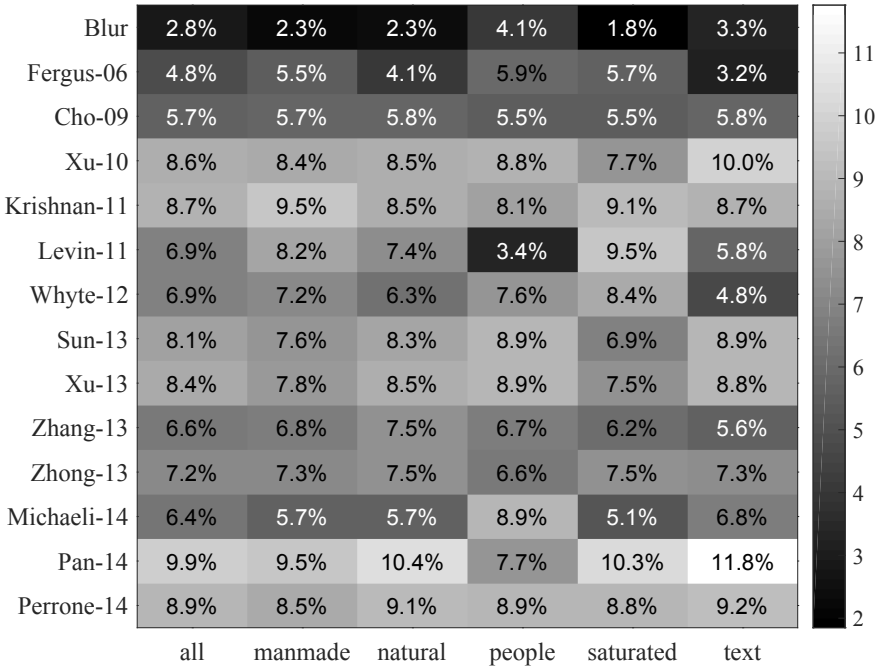


Figure 15: Percentage of the obtained votes with different attributes on our synthetic non-uniform dataset.

6. Rationale Analysis

For each image pair, we ask subjects to choose reason(s) why they prefer the selected image. Table 2 lists the reasons available to select when comparing a pair of images. We present the percentage of chosen reasons for all the datasets in Figure 16, 17, and 18.

ID	Reason
1	This image is sharper/clearer.
2	This image has less ringings.
3	This image has less noise.
4	This image has less saturated regions.
5	This image is NOT over-sharpen.
6	No specific reason. This image is simply more appealing.
7	I can't tell the difference. / I don't know which one is better.

Table 2: Reasons for choosing a result.

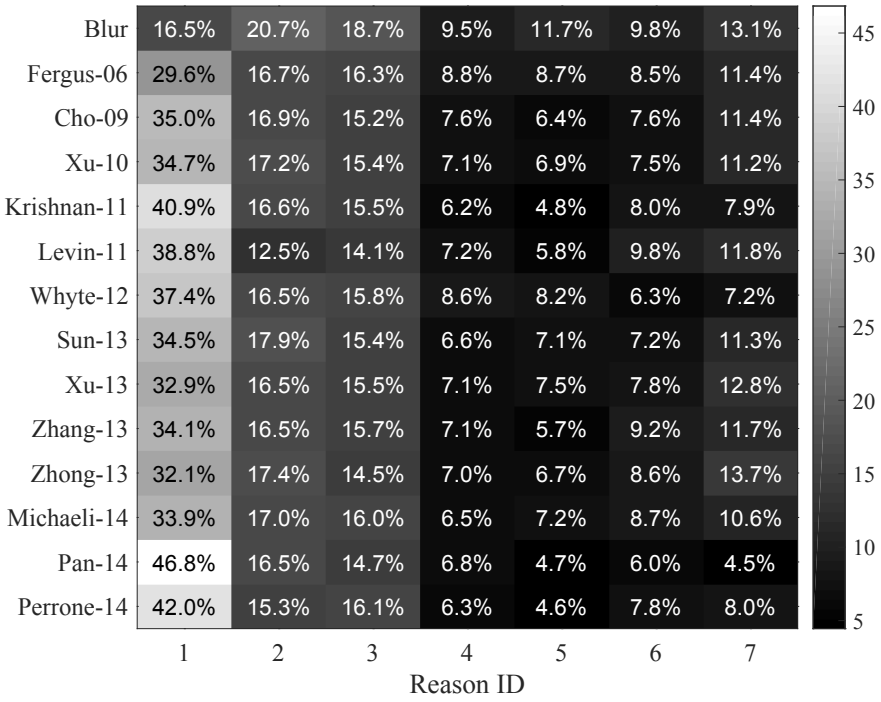


Figure 16: Percentage of chosen reasons for each method on our real dataset.

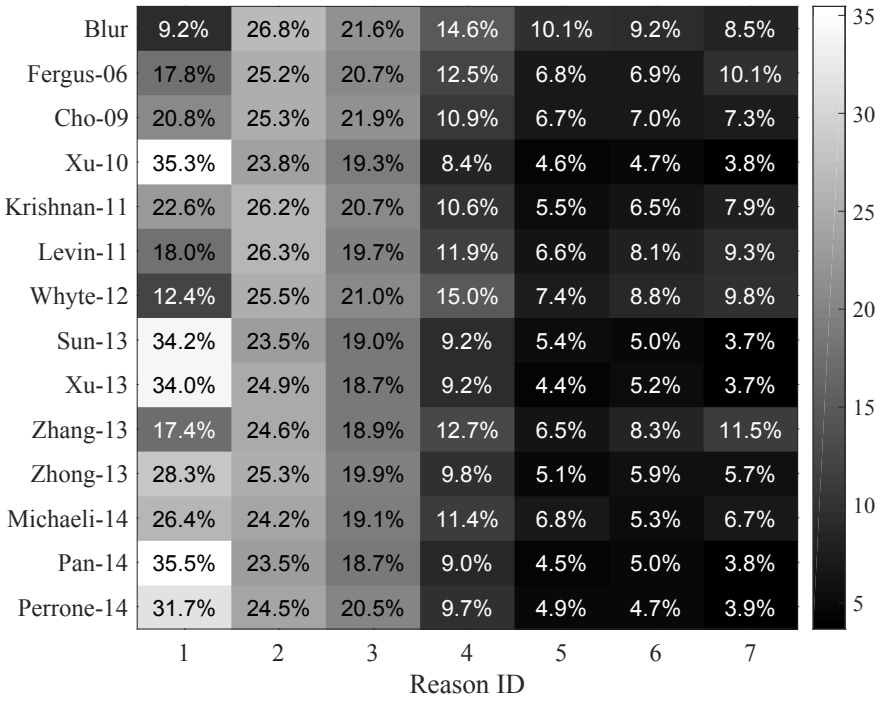


Figure 17: Percentage of chosen reasons for each method on our synthetic uniform dataset.

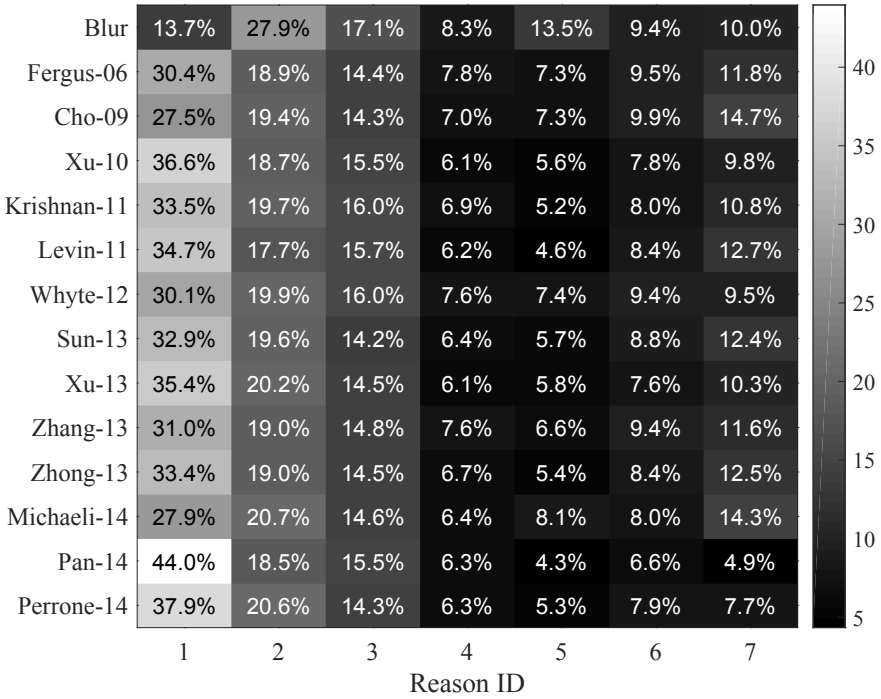


Figure 18: Percentage of chosen reasons for each method on our synthetic non-uniform dataset.

7. Deblurred Results

From Figure 19 to 28, we show 10 images from our real dataset to demonstrate the qualitative comparisons of the state-of-the-art deblurring algorithms.

(a) Blur, −0.8301

(b) Fergus-06 [3], −1.3698

(c) Cho-09 [2], 0.3213

(d) Xu-10 [12], 0.4522

(e) Krishnan-11 [4], 0.6420

(f) Levin-11 [5], −0.4208

(g) Whyte-12 [11], −0.0771

(h) Sun-13 [10], 0.1653

(i) Xu-13 [13], *NaN* [1]

(j) Zhang-13 [14], −0.0670

(k) Zhong-13 [15], −0.1584

(l) Michaeli-14 [6], 0.6087

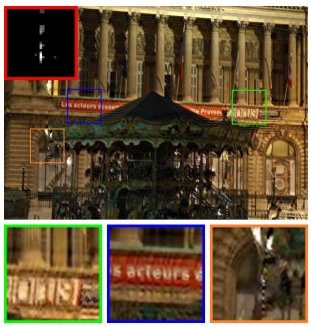
(m) Pan-14 [7], 0.6790

(n) Perrone-14 [8], 0.0546

Figure 19: Deblurred results from our real dataset and the corresponding B-T scores. Note that results with serious ringing artifacts are not favored by subjects.

[1] A method with *NaN* B-T score means that there is no subject prefers this method during the user study.

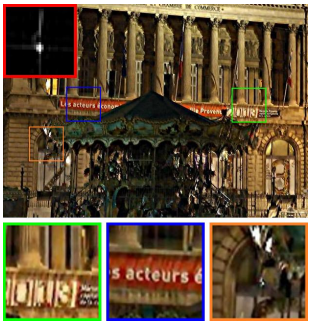
(a) Blur, −2.7480



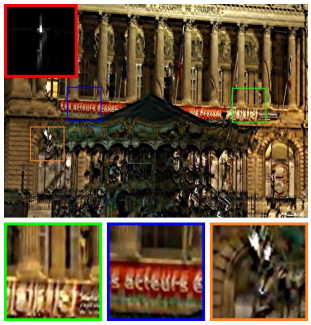
(b) Fergus-06 [3], −1.0439

(c) Cho-09 [2], 0.3110

(d) Xu-10 [12], 3.3246

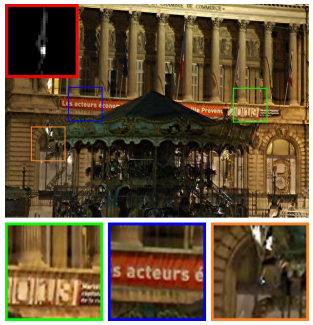


(e) Krishnan-11 [4], 1.4639



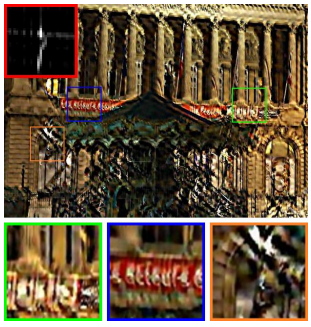
(f) Levin-11 [5], −1.7301

(g) Whyte-12 [11], 1.7181

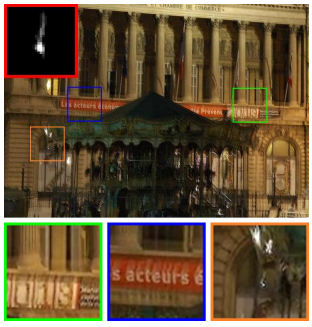


(h) Sun-13 [10], 2.1856

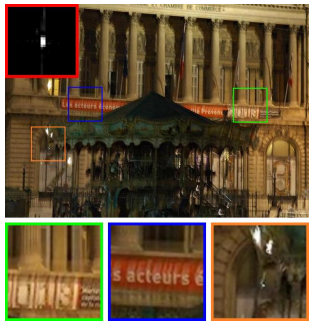
(i) Xu-13 [13], 2.2323



(j) Zhang-13 [14], −4.8348

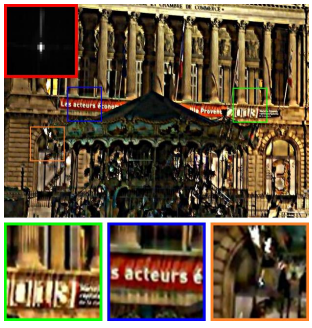


(k) Zhong-13 [15], −0.6232



(l) Michaeli-14 [6], −0.6019

(m) Pan-14 [7], 1.8060

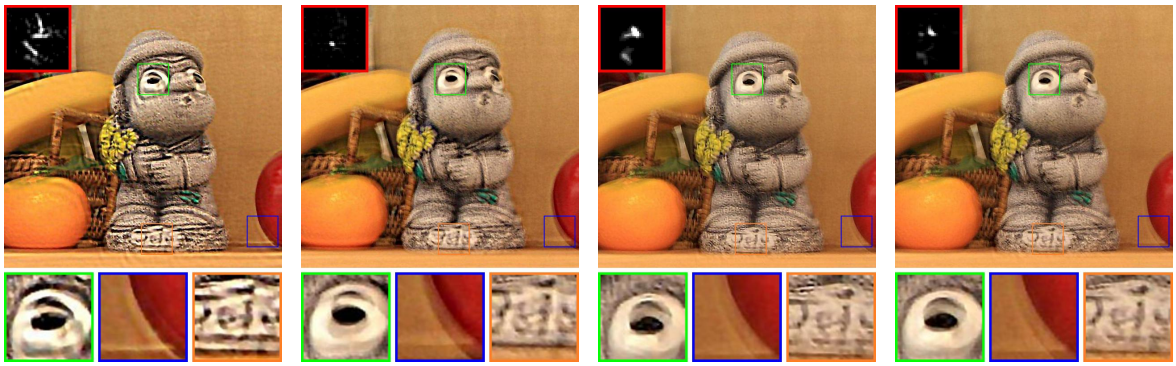


(n) Perrone-14 [8], −1.4596

Figure 20: An commonly used example in previous work with repeated structure, texts and saturated regions.

(a) Blur, −1.0501 (b) Fergus-06 [3], *NaN* (c) Cho-09 [2], 0.4211 (d) Xu-10 [12], 0.5418

(e) Krishnan-11 [4], 0.5656 (f) Levin-11 [5], −0.1513 (g) Whyte-12 [11], 1.1109 (h) Sun-13 [10], 0.6175



(i) Xu-13 [13], −0.5069 (j) Zhang-13 [14], −0.3559 (k) Zhong-13 [15], 0.1783 (l) Michaeli-14 [6], 0.3395

(m) Pan-14 [7], −1.5549 (n) Perrone-14 [8], −0.1556

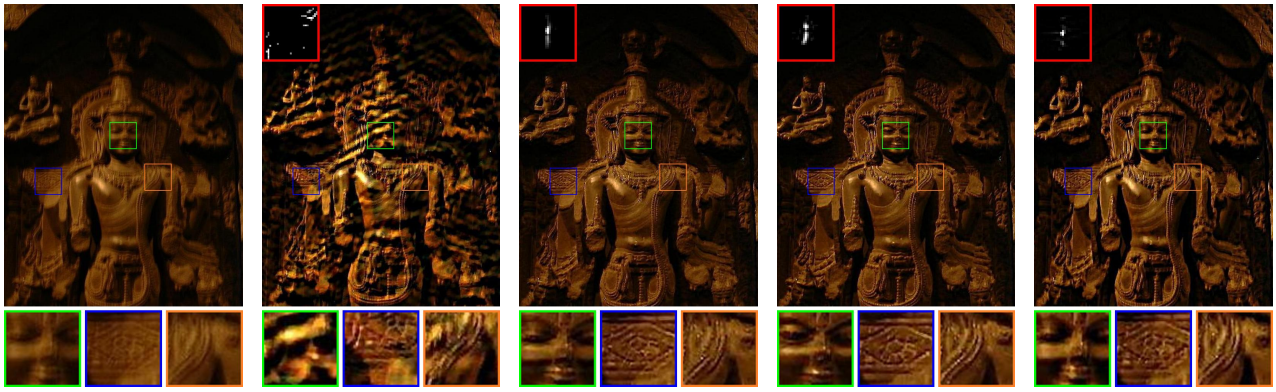
Figure 21: A challenging example. We note that most algorithms cannot recover the eyeball region well.

(a) Blur, −2.7592 (b) Fergus-06 [3], −2.7250 (c) Cho-09 [2], −0.7706 (d) Xu-10 [12], 2.4300 (e) Krishnan-11 [4], −0.778

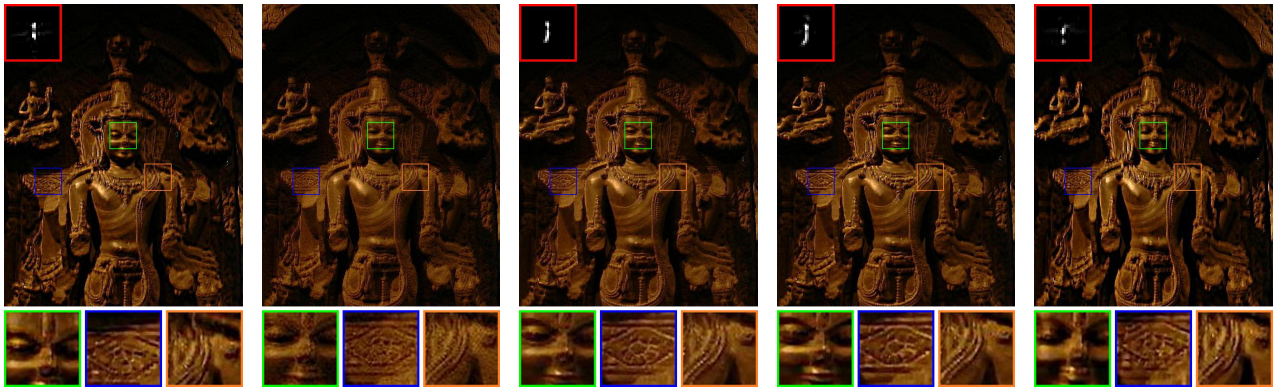
(f) Levin-11 [5], −3.0374 (g) Whyte-12 [11], −1.498 (h) Sun-13 [10], 2.7066 (i) Xu-13 [13], 2.1748 (j) Zhang-13 [14], −1.1505

(k) Zhong-13 [15], 1.4209 (l) Michaeli-14 [6], 0.1446 (m) Pan-14 [7], 3.4517 (n) Perrone-14 [8], 0.3912

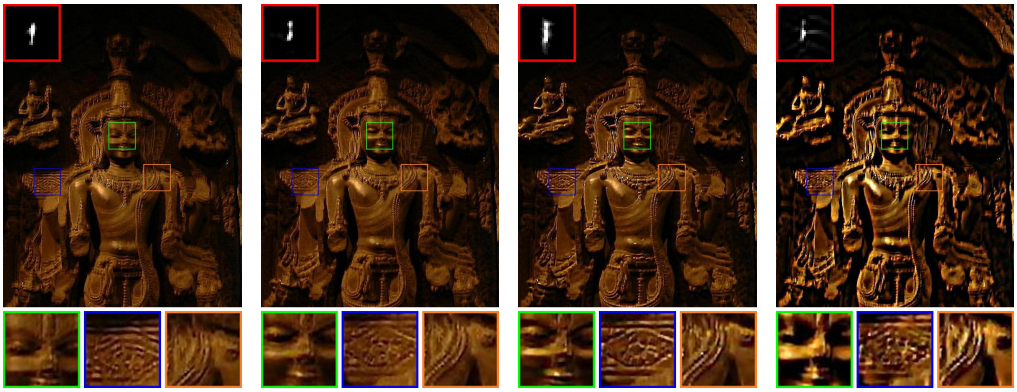
Figure 22: A commonly used example in previous.



(a) Blur, −1.4936 (b) Fergus-06 [3], *NaN* (c) Cho-09 [2], −0.2489 (d) Xu-10 [12], 0.8699 (e) Krishnan-11 [4], −0.272



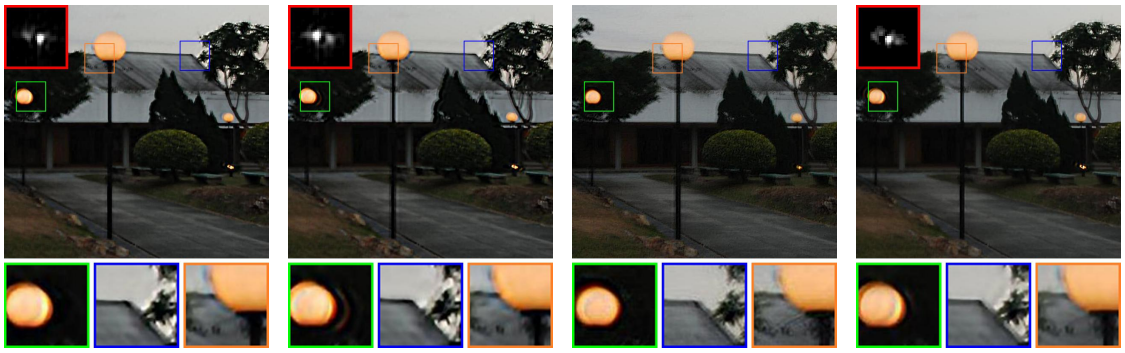
(f) Levin-11 [5], 0.2786 (g) Whyte-12 [11], −0.3159 (h) Sun-13 [10], 0.8026 (i) Xu-13 [13], 0.8303 (j) Zhang-13 [14], 0.1168



(k) Zhong-13 [15], −0.9789 (l) Michaeli-14 [6], 0.5529 (m) Pan-14 [7], 0.8574 (n) Perrone-14 [8], −0.9992

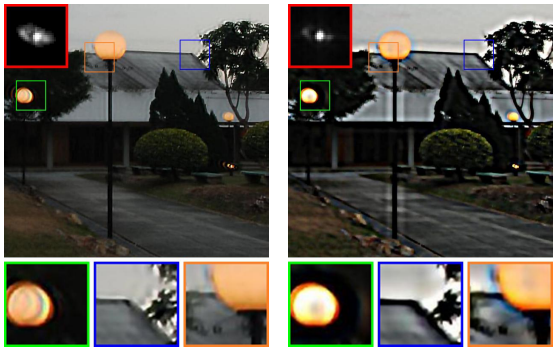
Figure 23: There is no clear winner in this example as the B-T scores of [7, 12, 13] are very close. Over-sharpened result [8] or the result with obvious ringing [3] are not favored by users.

(a) Blur, *NaN* (b) Fergus-06 [3], −0.5460 (c) Cho-09 [2], 0.6977 (d) Xu-10 [12], 0.3638



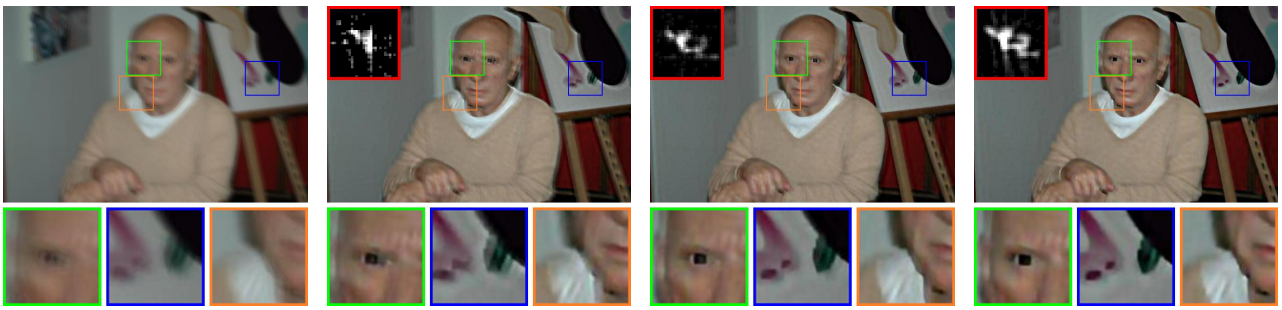
(e) Krishnan-11 [4], 0.6846 (f) Levin-11 [5], −0.1911 (g) Whyte-12 [11], 0.5656 (h) Sun-13 [10], 0.0381

(i) Xu-13 [13], 0.3581 (j) Zhang-13 [14], −0.9539 (k) Zhong-13 [15], −0.2357 (l) Michaeli-14 [6], 0.6029

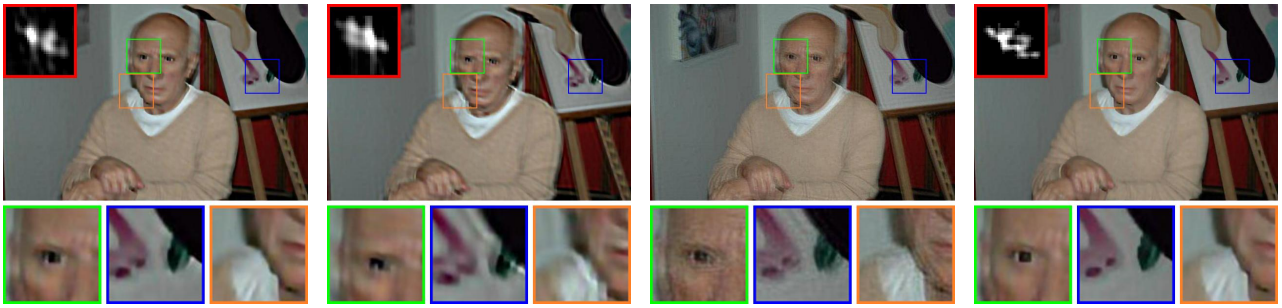


(m) Pan-14 [7], 0.0458 (n) Perrone-14 [8], −1.4299

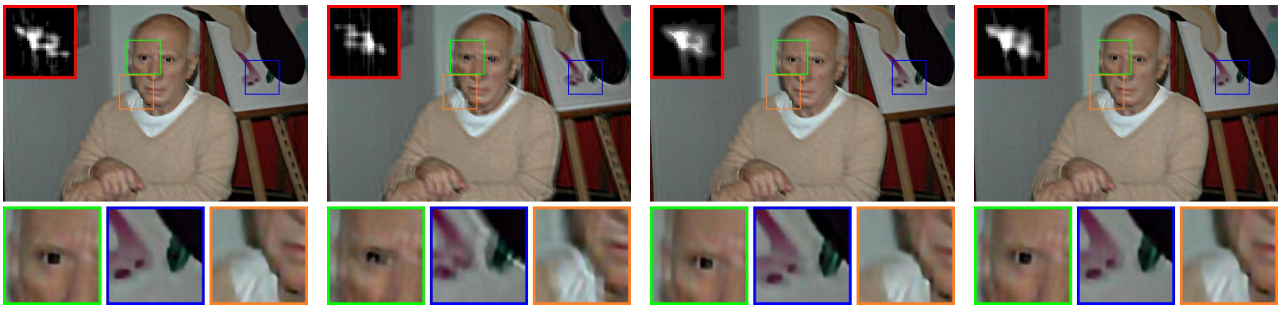
Figure 24: An example with large scene depth variations. Most algorithms produce visiable ringing artifacts around the lamp post.



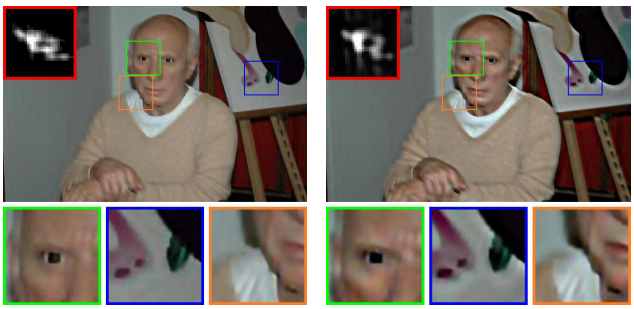
(a) Blur, −4.2615 (b) Fergus-06 [3], −1.9513 (c) Cho-09 [2], 2.2656 (d) Xu-10 [12], 1.5771



(e) Krishnan-11 [4], −1.0608 (f) Levin-11 [5], −0.9856 (g) Whyte-12 [11], −2.4611 (h) Sun-13 [10], 2.0881

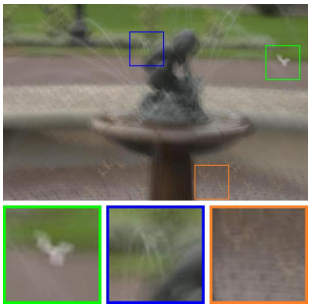


(i) Xu-13 [13], 2.0890 (j) Zhang-13 [14], −2.1276 (k) Zhong-13 [15], −0.3213 (l) Michaeli-14 [6], 1.9761

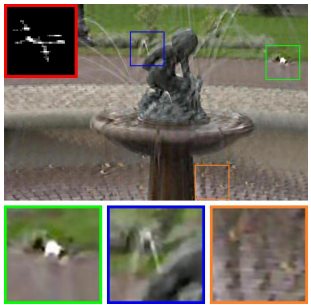


(m) Pan-14 [7], 2.0887 (n) Perrone-14 [8], 1.0846

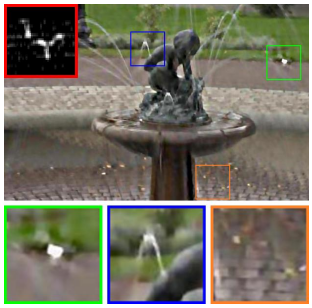
Figure 25: A commonly used example. It is challenging to accurately estimate the blur kernel with a spiral shape.



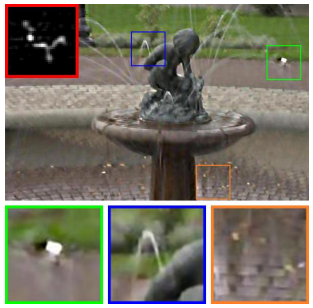
(a) Blur, −2.6297



(b) Fergus-06 [3], −1.0441

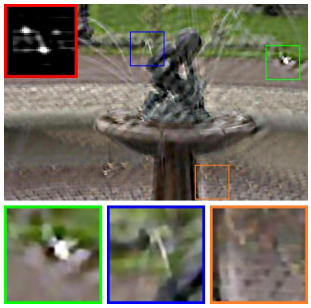


(c) Cho-09 [2], 1.5778

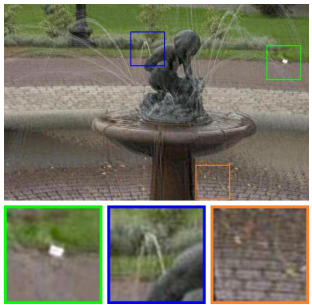


(d) Xu-10 [12], 1.2228

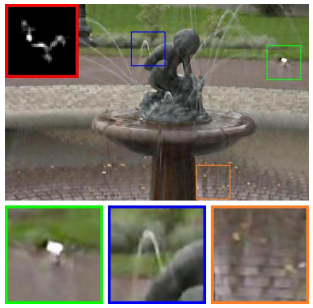
(e) Krishnan-11 [4], −0.0042



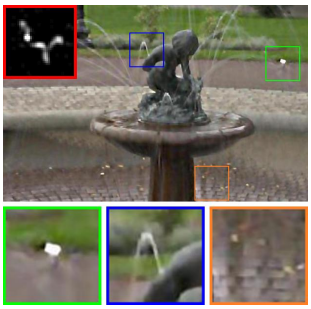
(f) Levin-11 [5], −2.0103



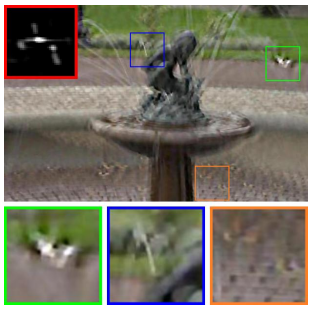
(g) Whyte-12 [11], 1.6796



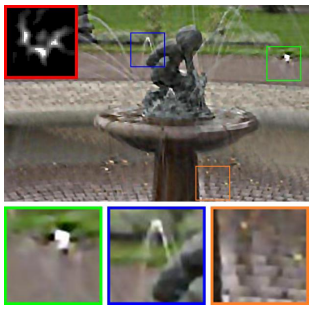
(h) Sun-13 [10], 1.4839



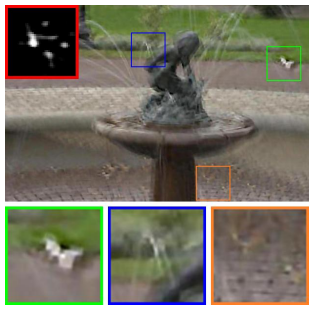
(i) Xu-13 [13], 1.3183



(j) Zhang-13 [14], −0.9308

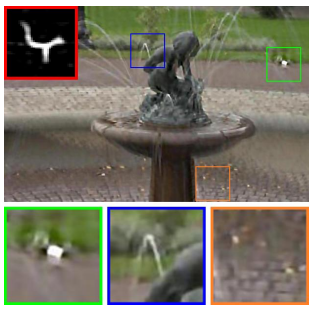


(k) Zhong-13 [15], 0.7685



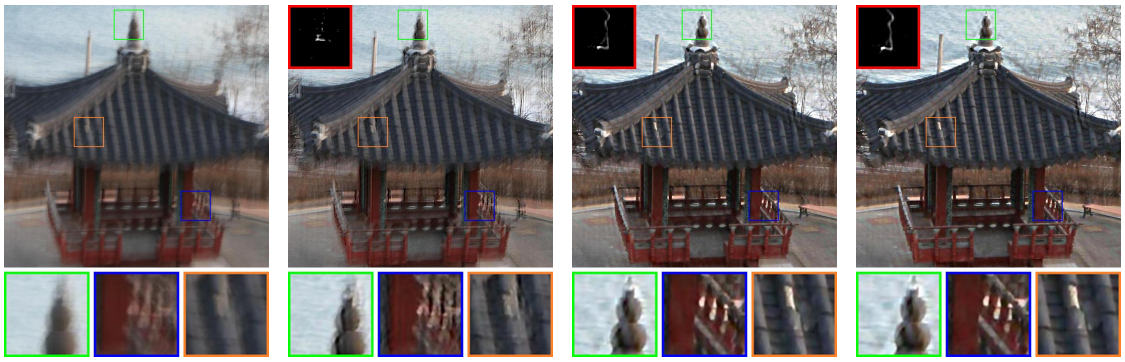
(l) Michaeli-14 [6], −1.7326

(m) Pan-14 [7], −0.1561

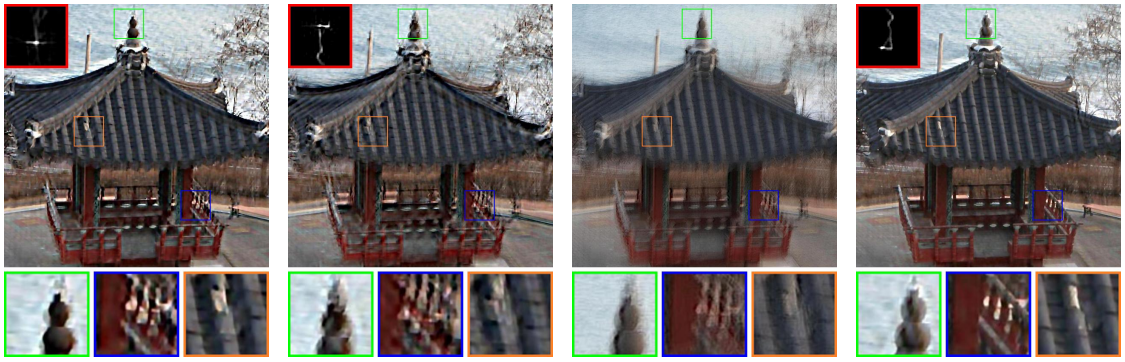


(n) Perrone-14 [8], 0.4569

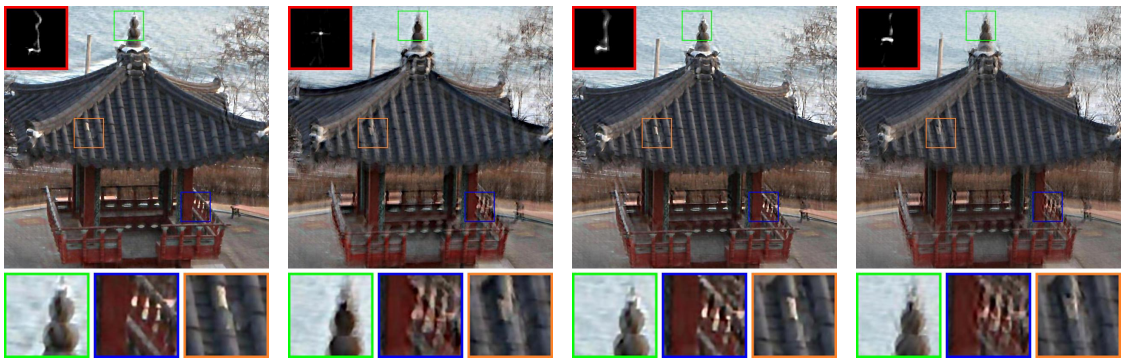
Figure 26: A commonly used example. Note that the green bounding box in (a) reveals the shape of the blur kernel.



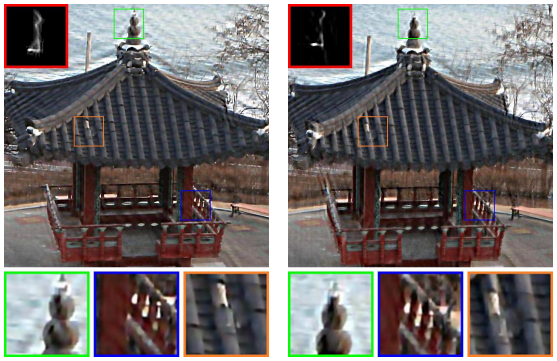
(a) Blur, −2.8083 (b) Fergus-06 [3], −1.0591 (c) Cho-09 [2], 1.1516 (d) Xu-10 [12], 2.2550



(e) Krishnan-11 [4], 0.1839 (f) Levin-11 [5], −2.4538 (g) Whyte-12 [11], −2.3836 (h) Sun-13 [10], 2.7478



(i) Xu-13 [13], 2.0553 (j) Zhang-13 [14], −1.5643 (k) Zhong-13 [15], 1.1471 (l) Michaeli-14 [6], −0.9064

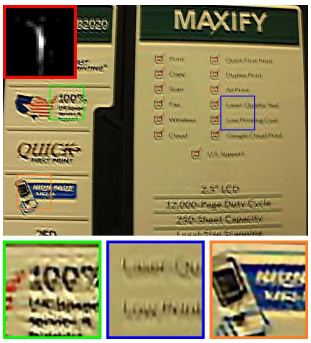


(m) Pan-14 [7], 0.3125 (n) Perrone-14 [8], 1.3222

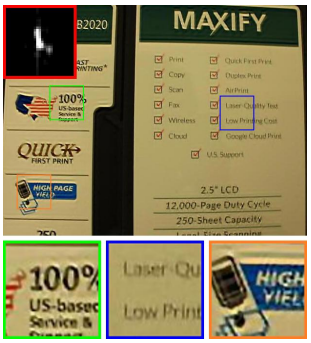
Figure 27: A commonly used example. Edge-based algorithms [10, 12] perform well in this example.

(a) Blur, −0.4624

(b) Fergus-06 [3], *NaN*

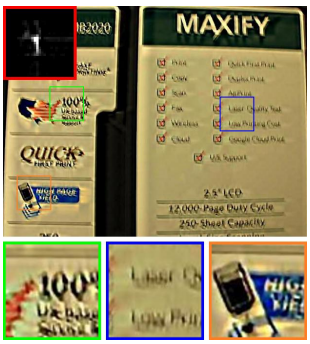


(c) Cho-09 [2], −1.3122



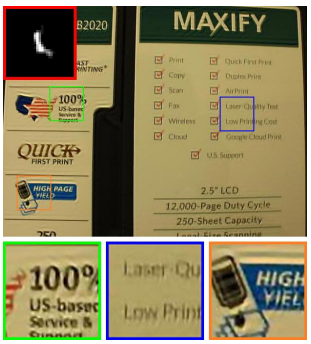
(d) Xu-10 [12], 0.9279

(e) Krishnan-11 [4], −0.3828

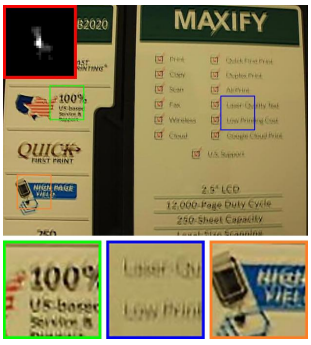


(f) Levin-11 [5], −1.2821

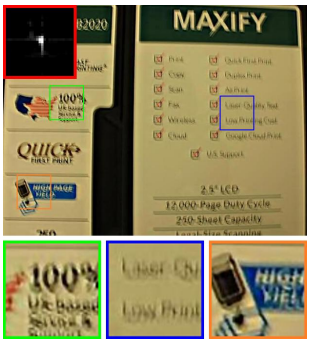
(g) Whyte-12 [11], 0.3157



(h) Sun-13 [10], 0.7890

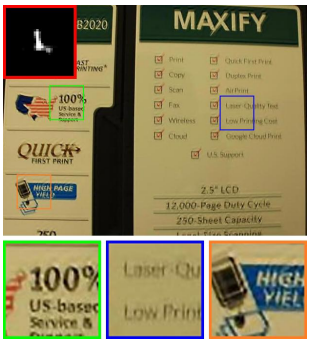


(i) Xu-13 [13], 0.2354

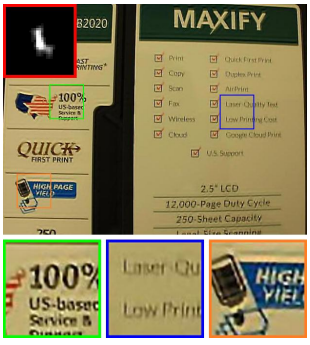


(j) Zhang-13 [14], −0.3735

(k) Zhong-13 [15], 0.1483



(l) Michaeli-14 [6], 0.6089



(m) Pan-14 [7], 1.0524

(n) Perrone-14 [8], −0.2646

Figure 28: An example with rich texts. Most algorithms produce noticeable ringing artifacts.

References

[1] R. A. Bradley and M. E. Terry. Rank analysis of incomplete block designs the method of paired comparisons. *Biometrika*, 39(3-4):324–345, 1952. 6

[2] S. Cho and S. Lee. Fast motion deblurring. *ACM TOG (Proc. SIGGRAPH Asia)*, 28(5):145:1–145:8, 2009. 3, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23

[3] R. Fergus, B. Singh, A. Hertzmann, S. T. Roweis, and W. T. Freeman. Removing camera shake from a single photograph. *ACM TOG (Proc. SIGGRAPH)*, 25(3):787–794, 2006. 3, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23

[4] D. Krishnan, T. Tay, and R. Fergus. Blind deconvolution using a normalized sparsity measure. In *CVPR*, 2011. 3, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23

[5] A. Levin, Y. Weiss, F. Durand, and W. T. Freeman. Efficient marginal likelihood optimization in blind deconvolution. In *CVPR*, 2011. 3, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23

[6] T. Michaeli and M. Irani. Blind deblurring using internal patch recurrence. In *ECCV*. 2014. 3, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23

[7] J. Pan, Z. Hu, Z. Su, and M.-H. Yang. Deblurring text images via l0-regularized intensity and gradient prior. In *CVPR*, 2014. 3, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23

[8] D. Perrone and P. Favaro. Total variation blind deconvolution: The devil is in the details. In *CVPR*, 2014. 3, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23

[9] U. Schmidt, C. Rother, S. Nowozin, J. Jancsary, and S. Roth. Discriminative non-blind deblurring. In *CVPR*, 2013. 2

[10] L. Sun, S. Cho, J. Wang, and J. Hays. Edge-based blur kernel estimation using patch priors. In *ICCP*, 2013. 3, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23

[11] O. Whyte, J. Sivic, A. Zisserman, and J. Ponce. Non-uniform deblurring for shaken images. *IJCV*, 98(2):168–186, 2012. 3, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23

[12] L. Xu and J. Jia. Two-phase kernel estimation for robust motion deblurring. In *ECCV*, 2010. 3, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23

[13] L. Xu, S. Zheng, and J. Jia. Unnatural L_0 sparse representation for natural image deblurring. In *CVPR*, 2013. 3, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23

[14] H. Zhang, D. Wipf, and Y. Zhang. Multi-image blind deblurring using a coupled adaptive sparse prior. In *CVPR*, 2013. 3, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23

[15] L. Zhong, S. Cho, D. Metaxas, S. Paris, and J. Wang. Handling noise in single image deblurring using directional filters. In *CVPR*, 2013. 3, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23