

Blind Image Deblurring with FFT-ReLU Sparsity Prior

Abdul Mohaimen Al Radi, Prothito Shovon Majumder, Md. Mosaddek Khan
University of Dhaka

abdulmohaimenal-2018925300@cs.du.ac.bd, prothitoshovon-2018725302@cs.du.ac.bd,
mosaddek@du.ac.bd

Abstract

*Blind image deblurring is the process of recovering a sharp image from a blurred one without prior knowledge about the blur kernel. It is a small data problem, since the key challenge lies in estimating the unknown degrees of blur from a single image or limited data, instead of learning from large datasets. The solution depends heavily on developing algorithms that effectively model the image degradation process. We introduce a method that leverages a prior which targets the blur kernel to achieve effective deblurring across a wide range of image types. In our extensive empirical analysis, our algorithm achieves results that are competitive with the state-of-the-art blind image deblurring algorithms, and it offers up to two times faster inference, making it a highly efficient solution.*¹

1. Introduction

Blind image deblurring aims to recover a sharp image I from a blurred image B without knowing the blur kernel k . Mathematically, the blurred image B is modeled as:

$$B = I \otimes k + \eta$$

where \otimes denotes convolution and η represents noise. The task is to estimate both I and k from B . This is formulated as an optimization problem:

$$\min_{I,k} \|B - I \otimes k\|^2 + \lambda \mathcal{R}(I, k)$$

where $\|\cdot\|^2$ measures the difference between the observed and predicted images, $\mathcal{R}(I, k)$ is a regularization term to enforce constraints, and λ is a regularization parameter. The goal is to recover both I and k by balancing data fidelity and regularization.

Blind image deblurring addresses a fundamental challenge in image processing: restoring clarity from blurred images when the exact nature of the blur is unknown. The

algorithmic approach generalizes the image deblurring process, helping data-driven methods perform better.

A common approach in blind image deblurring is the maximum a posteriori (MAP) framework, where the latent image and the blur kernel are alternately optimized. Various heuristics involving the latent image, the kernel, or both have been developed in this framework [3, 10, 20, 23, 32, 35, 48]. This results in new priors that produce effective outcomes. These methods often target specific types of images, as they rely on image properties that aid optimization [32]. Additionally, the computation of some priors can be resource-intensive [4, 33], which can affect the overall efficiency of deblurring techniques. While these approaches demonstrate promising results in their respective domains, their performance can deteriorate when applied to broader image categories, leading to inconsistent outcomes in some cases.

Mao *et al.* [28] show that applying a specific sequence of operations—removing negative frequencies from a blurry image, reconstructing it in the spatial domain, and subtracting half of the original image—yields an implicit blur kernel representation. This process helps in understanding the blur’s direction and intensity. Building on this insight, we discovered that, when applied to a sharp image, this sequence results in a null kernel, characterized by significantly fewer non-zero elements compared to any blur kernel’s implicit representation.

Our thorough empirical analysis demonstrates that the L_0 norm obtained after applying this sequence of operations to a blurry image consistently exceeds that of its sharp counterpart. Leveraging this property, we incorporated it into our optimization model, leading to a highly efficient algorithm. The linear nature of our calculation allows our algorithm to be up to two times faster than popular blind image deblurring methods, while still delivering competitive results in terms of PSNR, SSIM, and error ratio.

Our main contributions are summarized as follows:

1. We propose a new prior, the ReLU Sparsity prior,

¹The source code will be available in the authors’ Github profile.



Figure 1. Results of our blind image deblurring algorithm, compared with other state-of-the-art algorithms. From left to right: 1) Input blurry image, followed by results from 2) Chen *et al.* [4] 3) Wen *et al.* [42] 4) Pan *et al.* [32], 5) Pan *et al.* [33] and 6) our algorithm.

which allows us to obtain faithful information about the blur kernel regardless of the nature of the image, while simultaneously applying optimization techniques based on the change in sparsity that occurs due to convolution. Fig. 1 shows an example of deblurred output from our algorithm.

2. Our algorithm achieves competitive performance based on the PSNR, SSIM, and error ratio metrics when compared with the state-of-the-art blind image deblurring algorithms [9, 10, 20, 33–35, 44] while achieving significantly lower inference time on high-resolution images when compared using images from benchmark datasets [18, 22, 38].

In the following sections, we begin by discussing related works pertinent to our study, followed by a detailed explanation of our proposed prior and the empirical evidence supporting the generalizability of the novel property. We then present our experimental results, starting with quantitative analysis and concluding with qualitative evaluations.

2. Related Works

Due to the joint estimation of the blur kernel and the latent sharp image from the input blurry image, single-image deblurring is an under-constrained problem, which is typically formulated in MAP frameworks [27]. Initial advances relied on statistical priors and salient edge detection [9, 10, 20, 23, 35, 44, 48], although the existence of strong edges in latent images is not guaranteed.

Different image priors and likelihood estimations have been proposed to improve the efficiency of MAP frameworks. Such regularizations or assumptions about the blur kernel or the latent image include L_0 regularised prior [32, 47], dark channel prior [33], extreme channel prior [49], patch prior [26], local binary pattern prior [51], latent structure prior [2], learned image prior [24], uniform blur [22, 48], non-uniform blur with multiple homographies [14, 30], local maximum difference prior [25], internal patch recurrence [29], Laplacian prior [7], tri-segment intensity prior [50], graph-based image prior [1], constant depth [11, 45], in-plane rotation [37], and forward motion

[52]. Fergus *et al.* [10] use variational Bayesian inference to learn an image gradient prior. Levin *et al.* [22] show that this method can avoid trivial solutions that naive MAP-based methods can possibly be unable to.

Sparsity priors have been shown to be useful in kernel estimation in MAP frameworks. Krishnan *et al.* [20] employed a normalized sparsity prior in their MAP framework for kernel estimation. Xu and Jia [44] introduced a two-phase technique for single-image deblurring. First, they estimated the blur kernel using edge selections and ISD optimizations. This was followed by non-blind deconvolution with total variation and an added Gaussian prior [9, 16].

Some comparatively newer approaches designed for domain-specific deblurring, such as low-illumination [13], text [6, 8, 32] and face [31] images utilize inherent statistical characteristics of their domains. Chen *et al.* [4] propose the local maximum gradient (LMG) prior, which utilizes the diminishing of the maximum value of a local patch during the blurring process. Wen *et al.* [42] propose the patch-wise minimum pixel (PMP) prior, which is built on the fact that the local minimal pixel would have a higher value in intensity after the blurring process due to the smoothing effect on the image pixels of the process itself. Both these priors use non-linear optimization schemes in MAP frameworks and perform well on both natural and specific images.

3. FFT-ReLU Sparsity Prior

In this section, we discuss the sparsity property of applying RFT operation and develop the FFT-ReLU Sparsity prior to formulate an objective function, in order to estimate the latent sharp image and blur kernel.

3.1. Sparsity and RFT

Sparsity: To describe our work, we begin by discussing the effect of convolution on the sparsity of images. In standard deblurring models, blurry images are denoted as the convolution of the sharp image and a blur kernel. Essentially, the expression can be written as:

$$B(x, y) = \sum_{i=0}^{k-1} \sum_{j=0}^{k-1} S(x+i, y+j) \cdot K(i, j) \quad (1)$$

Here, (x, y) denotes the pixel location, and the blur kernel K has a size of $k \times k$. S and B are the sharp image and the blurry image, respectively.

Since the output of convolution essentially represents a locally weighted linear combination of the input, it is essentially less sparse (meaning it has more nonzero elements, due to the multiplicative and additive properties of convolution) than the input. Therefore, a blurry image has fewer pixels which are very low in brightness (i.e. it has more nonzero elements), compared to the corresponding sharp image.

RFT: As introduced by Mao *et al.* [28], RFT is defined as a function that computes the following for an image:

1. Computes the FFT of the input blurry image
2. Applies the ReLU activation function on it
3. Computes the inverse FFT
4. Subtracts half of the image from the result

Formally, we can describe RFT as:

$$RFT(I) = \mathcal{F}^{-1}(\text{ReLU}(\mathcal{F}(I))) - \frac{1}{2}I \quad (2)$$

Mao *et al.* show that faithful information about the blur pattern, such as the direction and the level of blur, can be discovered by applying this sequence of operations. We observe that applying RFT to a sharp image results in fewer non-zero elements than its blurry counterpart, which provides us with the opportunity to penalize the nonzero results from RFT since those indicate the presence of blur.

We apply the RFT operations on the images from various datasets, including one provided by Sun *et al.* [38], datasets of text images [36], low-light images [41], face images [15] and observe the values of the L_0 norm to empirically substantiate our observation in Fig. 2. Since only Sun *et al.* provided blurry images with their corresponding sharp pairs, we generated blurry images from the provided sharp images for the other datasets by using blur kernels provided by Chen *et al.* [5] randomly. We can see the change in L_0 norm in blurry images from its sharp counterpart when RFT operation is applied to both of them in Fig. 2.

3.2. Model and Optimization

Adding $RFT(I)$ to standard formulations of image deblurring, we express our objective function as follows:

$$\min_{I,k} \|I \otimes K - B\|_2^2 + \alpha \|k\|_2^2 + \beta \|\nabla I\|_0 + \lambda \|RFT(I)\|_0 \quad (3)$$

Here, the first term ensures that the given blurred image and the convolution of the estimated latent image and blur kernels are as similar as possible. The second term applies regularisation on the kernel. The third term retains large gradients and discards smaller ones [32, 48]. α, β and λ are penalty parameters. From Eq. (3), we solve alternately for

the I and k using coordinate descent using following equations respectively:

$$\min_I \|I \otimes k - B\|_2^2 + \beta \|\nabla I\|_0 + \lambda \|RFT(I)\|_0 \quad (4)$$

$$\min_k \|I \otimes k - B\|_2^2 + \alpha \|k\|_2^2 \quad (5)$$

Our algorithm can be partitioned into blind deconvolution and non-blind deconvolution segments. In blind deconvolution, the computation of the latent image and the blur kernel is done without an exact PSF by using priors or heuristics. The non-blind deconvolution removes the ringing artifacts that can appear from blind deconvolution.

3.3. Blind Deconvolution

The purpose of blind deconvolution is to estimate the sharp image I and the blur kernel k from the input image B .

Estimating I: The non-linear RFT function and L_0 regularisation make the minimization of Eq. (4) challenging in terms of computation. Half-quadratic splitting methods [46] are used to handle L_0 minimization. We split the gradients into $g = (g_x, g_y)$ for gradients across x and y-axis, and introduce auxiliary variable h for $RFT(I)$. Rewriting Eq. (4), we now have:

$$\min_{I,h,g} \|I \otimes k - B\|_2^2 + \gamma \|\nabla I - g\|_2^2 + \beta \|RFT(I) - h\|_2^2 + \mu \|g\|_0 + \lambda \|h\|_0 \quad (6)$$

where γ and β are penalty parameters. Solutions for individual variables (I, u , or g) can be calculated when the others are held constant. In order to use Eq. (6) in Algorithm 1 to estimate the latent sharp image, we transform this in the frequency domain and reach a closed-form solution in Eq. (8), Eq. (9), and Eq. (10).

In order to solve for the nonlinear $RFT(I)$, we observe that a linear operator F can be used as a substitute when applied to the I in vector form i.e. for the latent image, $FI = RFT(I)$. To compute the values of F , we use gradient descent with Adam optimizer [17]. Using repeated reconstruction, we iteratively compute F from the previous estimation of I . We look at all images from TextOCR dataset [36], LOL dataset [41], and LFW dataset [15] and observe that FI closely approximates $RFT(I)$ and converges reasonably within 100 steps in Figure Fig. 3.

Given F , we can solve for I from:

$$\min_I \|\mathbf{T}_k \mathbf{I} - \mathbf{B}\|_2^2 + \gamma \|\nabla \mathbf{I} - \mathbf{g}\|_2^2 + \beta \|\mathbf{FI} - \mathbf{h}\|_2^2 \quad (7)$$

where \mathbf{T}_k is a Toeplitz matrix of k , which is multiplied with vectors using FFT [39]. \mathbf{B}, \mathbf{g} and \mathbf{u} are denoted in their vector forms, respectively. The closed-form solution for I can

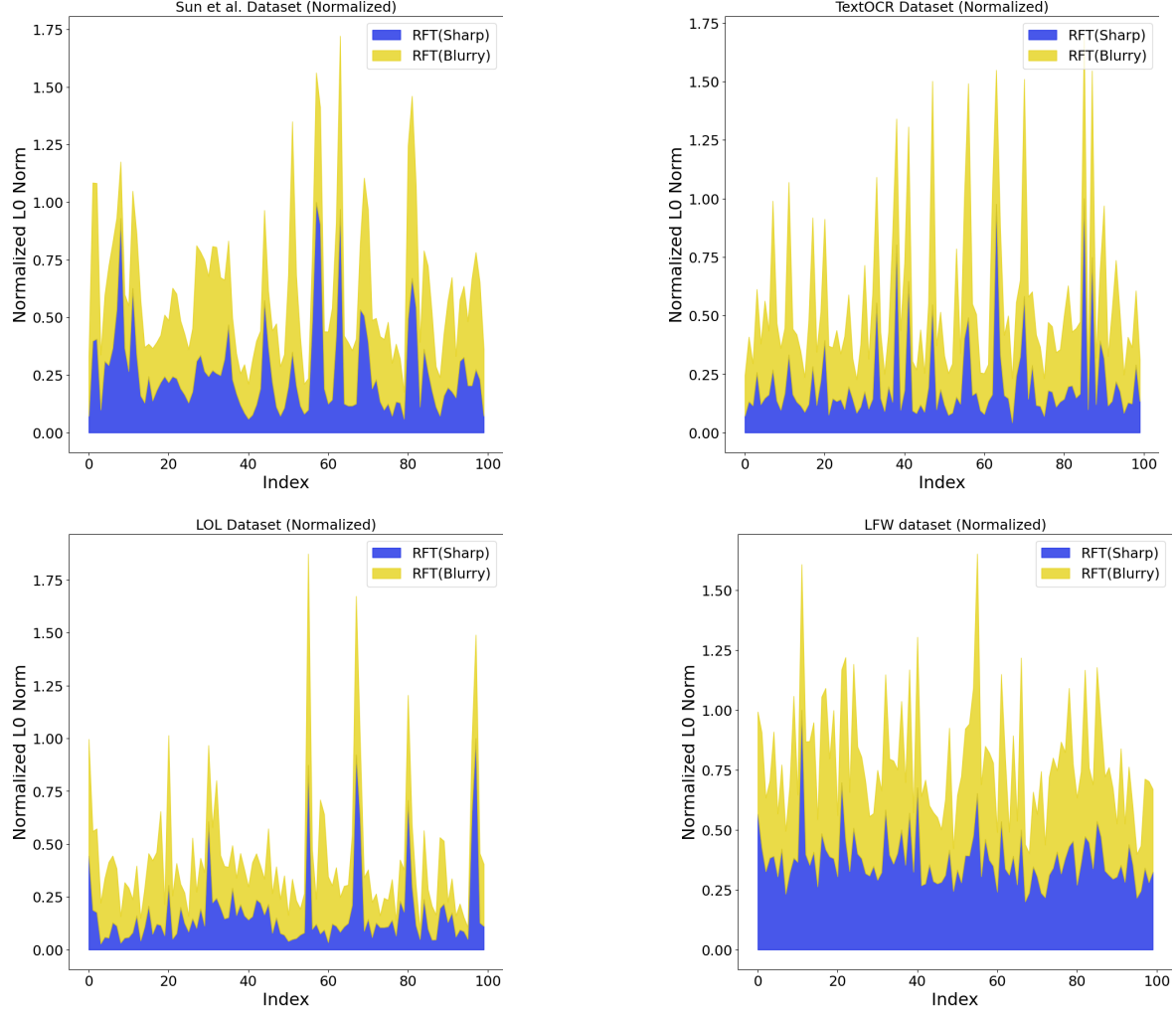


Figure 2. Decreased Sparsity in Blurry Images after RFT, as demonstrated in images from Sun *et al.* [38], TextOCR dataset, [36] LOL dataset, [41] and LFW dataset [15].

be derived as follows:

$$\begin{aligned} \mathbf{I} &= \frac{\mathbf{T}_k^T \mathbf{B} + \gamma \nabla^T \mathbf{g} + \beta \mathbf{F}^T \mathbf{h}}{\mathbf{T}_k^T \mathbf{T}_k + \gamma \nabla^T \nabla + \beta \mathbf{F}^T \mathbf{F}} \\ &= \mathcal{F}^{-1} \left(\frac{\overline{\mathcal{F}(\mathbf{K})} \mathcal{F}(\mathbf{B}) + \gamma \overline{\mathcal{F}(\nabla)} \mathcal{F}(\mathbf{g}) + \beta \overline{\mathcal{F}(\mathbf{F})} \mathcal{F}(\mathbf{h})}{\overline{\mathcal{F}(\mathbf{K})} \mathcal{F}(\mathbf{K}) + \gamma \overline{\mathcal{F}(\nabla)} \mathcal{F}(\nabla) + \beta \overline{\mathcal{F}(\mathbf{F})} \mathcal{F}(\mathbf{F})} \right) \end{aligned} \quad (8)$$

The detailed derivation of this solution is presented in the supplementary material. Given \mathbf{I} , we can solve for \mathbf{h} from:

$$\min_h \beta \|\mathbf{RFT}(\mathbf{I}) - \mathbf{h}\|_2^2 + \lambda \|\mathbf{h}\|_0 \quad (9)$$

and for \mathbf{g} from:

$$\min_g \gamma \|\nabla \mathbf{I} - \mathbf{g}\|_2^2 + \mu \|\mathbf{g}\|_0 \quad (10)$$

using element-wise minimization. The solution for \mathbf{h} can be

written as:

$$\mathbf{h} = \begin{cases} \mathbf{RFT}(\mathbf{I}), & |\mathbf{RFT}(\mathbf{I})|^2 \geq \frac{\lambda}{\beta} \\ 0, & \text{otherwise} \end{cases} \quad (11)$$

The solution for \mathbf{g} can also be written in a similar manner, using element-wise minimization techniques.

Estimating \mathbf{k} : Eq. (5) is a least squares problem when \mathbf{I} is given. Due to the proven accuracy of gradient-based methods for estimating kernels in [9, 23, 48], \mathbf{k} is computed from:

$$\min_k \|\nabla \mathbf{I} \otimes \mathbf{k} - \nabla \mathbf{B}\|_2^2 + \alpha \|\mathbf{k}\|_2^2 \quad (12)$$

FFTs are used for solving Eq. (12), using techniques mentioned in [9, 32, 48]. The negative elements of the kernel are set to zero and the obtained kernel is normalized.

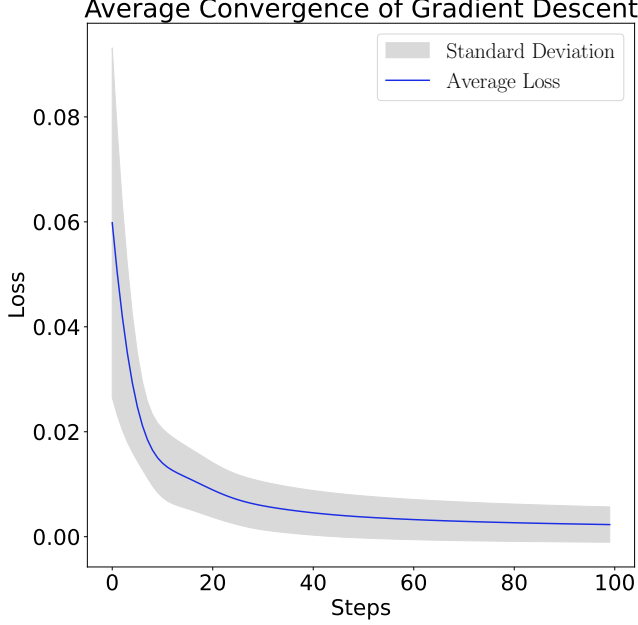


Figure 3. Using gradient descent, FI and RFT(I) converge reasonably within 100 steps.

Algorithm 1: Blind Deconvolution

Input : B : Blurry Image
Output: I : Latent Image (Intermediate), k : Blur Kernel

```

1  $k \leftarrow$  initialised from coarse resolution
2  $scale \leftarrow$  computed according to  $k$ 
3 for  $i = 0$  to  $max\_scale$  do
4    $B_{intermediate} \leftarrow$  downsample( $B, scale$ )
5   for  $j = 0$  to  $max\_iter$  do
6      $I \leftarrow$  solution of Equation (6)
7      $k \leftarrow$  solution of Equation (5)
8      $k \leftarrow$  remove_isolated_noise( $k$ )
9      $k \leftarrow$  adjust_psf_center( $k$ )
10  end
11   $k \leftarrow$  estimate_psf( $k, scale$ )
12   $scale \leftarrow scale + 1$ 
13 end
14 return  $I, k$ 
```

This blind deconvolution algorithm estimates the latent image I and the blur kernel k , which is necessary for obtaining the latent image. Initially, the kernel k is initialized to the coarsest resolution (Line 1). Line 2 determines the scale according to the current kernel size. The algorithm iteratively increases the scale. For each scale within a loop, the blurry image B is down-sampled to the current scale in Line 4. We iterate a predetermined number of times within this setting to refine the estimation of I using Equation (6) in Line 6, and k using Equation (5) in Line 7. Iso-

lated noises are removed by deleting connected components within the kernel that do not exceed a specified threshold (Line 8). The kernel is centered after the noise removal in Line 9, the kernel is then up-sampled with the current scale in Line 11. All down-sampling and up-sampling are done using bi-linear interpolation.

3.4. Non-blind Deconvolution

Algorithm 2: Non-blind Deconvolution

Input : I : Intermediate Latent Image, K : Intermediate Blur Kernel
Output: I : Intermediate Latent Image with Ringing Artifacts Removed

```

1 for  $i = 0$  to  $num\_channels$  do
2    $I_c \leftarrow$  laplacian_prior_estimation( $I_{channel_I}$ )
3    $I_1 \leftarrow$  concatenate( $I_1, I_c$ )
4 end
5  $I_2 \leftarrow$  solution for  $I$  using equation Eq. (4)
6  $diff \leftarrow I_1 - I_2$ 
7  $I\_result \leftarrow I_1 -$  bilateral_filter( $diff$ )
8 return  $I\_result$ 
```

For non-blind deconvolution, methods based on Laplacian priors [19] tend to be effective in preserving fine spatial details. We estimate an image I_1 with such priors in Line 2 and Line 3. Then, we use the L_0 norm of the gradient from our equation Eq. (4) to estimate another latent image I_2 in Line 5, since gradient-based methods have been shown to provide accurate results in suppressing ringing artifacts. Similar to ringing suppression methods discussed in [35], we compute a difference map between these two estimated images in Line 6. We use bilateral filtering on the computed difference map and subtract the result from I_1 in Line 7, thereby smoothing out artifacts and completing non-blind deconvolution.

4. Experiment

In this section, we discuss the parameter settings, quantitative results on datasets [18, 22], and qualitative results on images from different distributions.

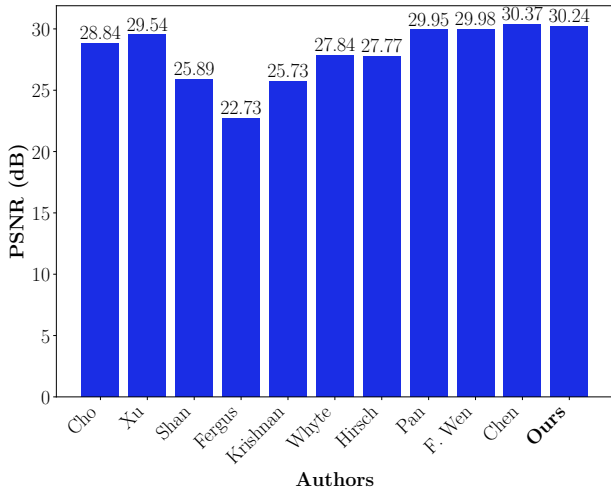
4.1. Empirical Setting

The parameters λ, μ and α are assigned values $3 \times 10^{-4}, 0.004$ and 2, respectively. We compared the convergence of the algorithm for various values of these parameters with respect to kernel similarity (defined as SSIM between the estimated kernel and ground-truth kernel), and a more detailed analysis of this is provided in the supplementary material. The value of max_iter is set to be 5 as a trade-off between timing and precision. The non-blind deconvolution techniques mentioned by Pan *et al.* [32] are followed. We provide a Python codebase for blind image

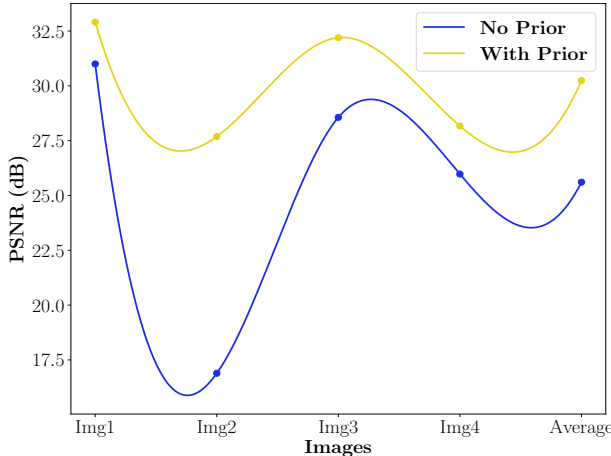
deblurring on single images. In order to ensure fairness in comparisons with pre-existing algorithms that were not implemented using Python, we avoided using GPU acceleration. However, since the previous works on blind image deblurring were implemented primarily on MATLAB, the translation to Python offers the opportunity for GPU access in the future as well as extension with deep learning methodologies.

4.2. Quantitative Results

In this section, we evaluate our proposed algorithm with the state-of-the-art algorithms based on the metrics of PSNR, SSIM [40], and error ratio. Datasets from Köhler *et al.* [18] and Levin *et al.* [22] are widely-used benchmarks, and are used for evaluating our work against other blind image deblurring algorithms.



(a) PSNR comparison with other algorithms



(b) PSNR with vs without our prior

Figure 4. PSNR comparison for Köhler *et al.* dataset

PSNR on Köhler *et al.* dataset: There are a total of 32 images. For each of the 4 images and 8 kernels,

there are 199 corresponding ground truth images for computing PSNR. We calculated the results of our algorithm for all of these images and plotted the average for each of the 4 images and also for the average of the 4 images against other statistical deblurring algorithms from [9, 10, 12, 20, 33, 35, 43, 48] in Fig. 4a. We also presented a comparison of PSNR results when our prior is used versus when it is not used in Fig. 4b.

Levin *et al.* dataset: This dataset also contains 4 images and 8 kernels each. The images are grayscale and have dimensions of 255×255 . We compare our PSNR, SSIM, error ratio, and inference times with the works in [4, 9, 33, 34, 42].

	PSNR	SSIM	Error ratio	Average inference time (seconds)
[33]	27.54	0.8626	1.2076	109.6088
[34]	28.38	0.925	0.8776	15.0949
[42]	26.1235	0.83637	1.48	18.61
[4]	26.48693	0.8515	1.01234	65.2
Ours	28.34	0.887	1.78	8.057

Table 1. Comparison on Levin *et al.* dataset

In Tab. 1, we show that our method achieves the best inference time while performing competitively in terms of PSNR, SSIM, and inference time.

Inference Time Comparison: We compare our work with [33] and [34] in this section, using datasets from Levin *et al.* [22], Köhler *et al.* [18] and Sun *et al.* [38], respectively. We report the inference times along with the image dimensions in the table below.

Algorithm	Inference Time (seconds)		
	255×255	800×800	1024×800
Pan <i>et al.</i> [33]	109.609	1550.932	2280.716
Pan <i>et al.</i> [34]	15.095	210.981	269.872
Wen <i>et al.</i> [42]	18.61	54.416	66.134
Chen <i>et al.</i> [4]	65.2	755.43	794.186
Ours	8.057	39.5764	55.72

Table 2. Inference Times of Algorithms on Images of Different Dimensions

Our approach achieves significant speed gains: nearly two times lower compared to [34], as presented in Tab. 2.

4.3. Visual Results

In this section, we present the results of our algorithm on blurry images from different domains, which we compare with other state-of-the-art blind image deblurring algorithms. The images are taken from datasets provided by Lai *et al.* [21], Levin *et al.* [22] and Köhler *et al.* [18].

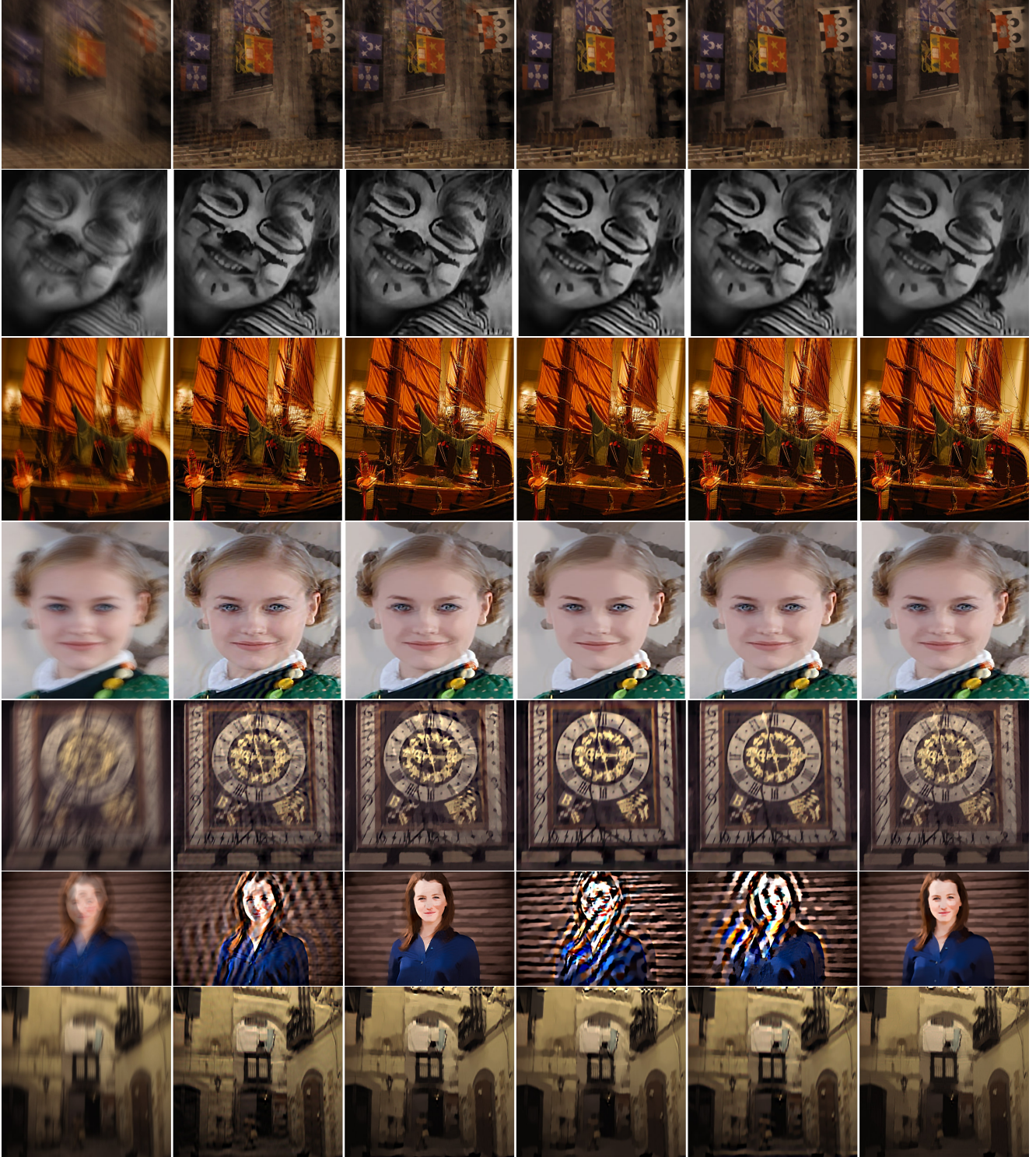


Figure 5. Results of our blind image deblurring algorithm, compared with other state-of-the-art algorithms. From left to right: 1) Input blurry image, followed by results from 2) Chen *et al.* [4] 3) Wen *et al.* [42] 4) Pan *et al.* [32], 5) Pan *et al.* [33] and 6) our algorithm.

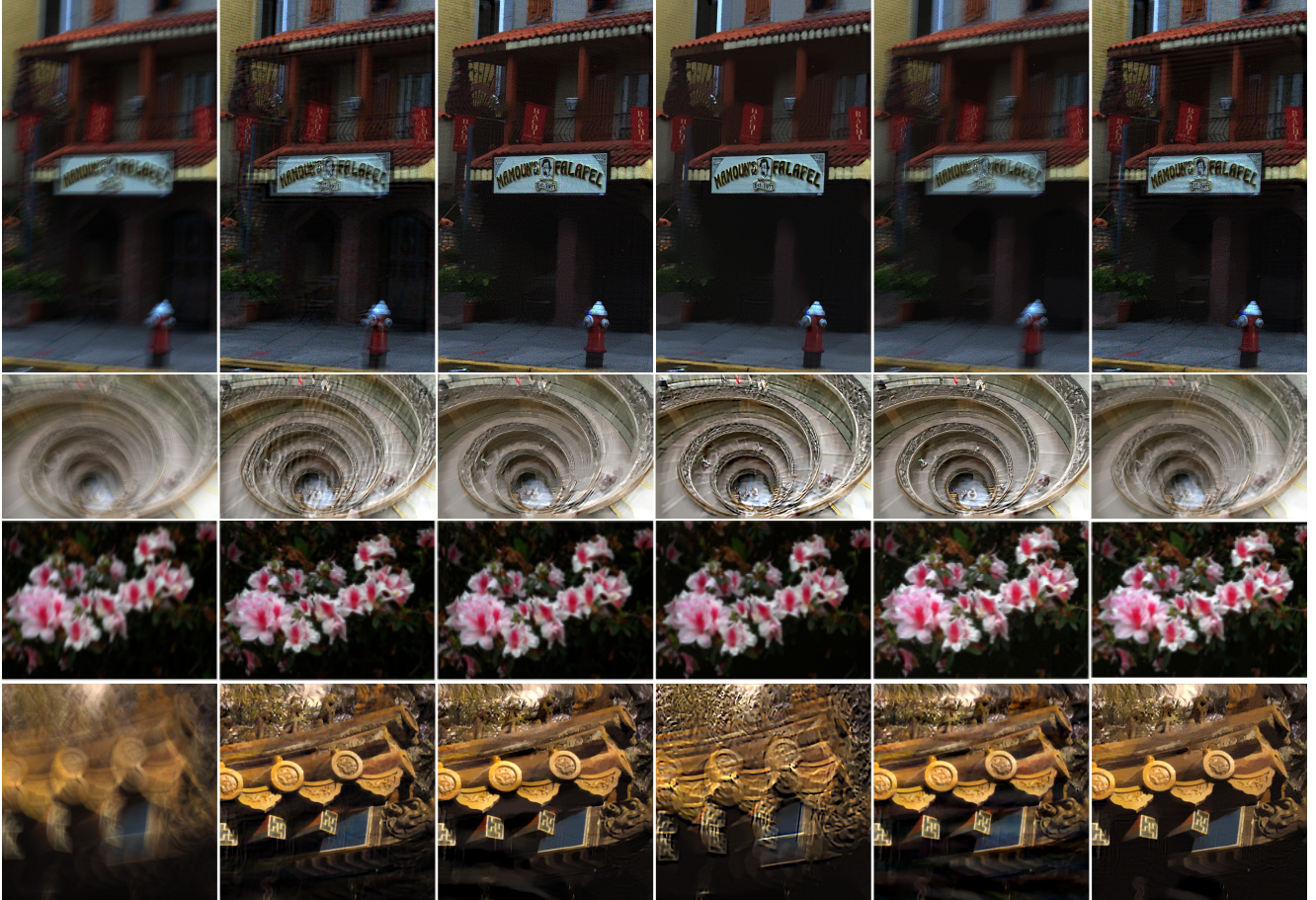


Figure 6. More results of our blind image deblurring algorithm, compared with other state-of-the-art algorithms. From left to right: 1) Input blurry image, followed by results from 2) Chen *et al.* [4] 3) Wen *et al.* [42] 4) Pan *et al.* [32], 5) Pan *et al.* [33] and 6) our algorithm.

We observe that our algorithm provides competitive results in terms of visual appeal, as well as qualitative measures mentioned in the previous section, for images irrespective of the distribution or domain they are taken from.

5. Conclusions and Remarks

In this paper, we propose a new prior for blind image deblurring based on observations made on the effects of convolution on the sparsity of an image. We made use of Fourier transforms in order to obtain the frequency components in order to separate the negative phase(s), which enabled us to obtain the blur kernel implicitly through information on blur level and blur direction. The ReLU Sparsity prior can be used to penalize a higher L_0 norm, which is an indicator of the blurriness of an image. For the non-convex optimization problem, we use half-quadratic splitting strategies. Our algorithm provides competitive performance on qualitative metrics such as PSNR, SSIM, and error ratio while providing up to two times lower inference times than the established state-of-the-art blind deblurring

algorithms.

Future work: We plan to explore the integration of our algorithm’s rapid inference and portable framework with deep learning architectures to further enhance deblurring evaluations. Additionally, we aim to develop alternative mathematical strategies to reduce the computational demands associated with gradient descent in RFT image computation. Another area of focus will be leveraging inherent image attributes such as saturation, intensity, and illumination to optimize the estimation of multiple kernels, thereby improving the recovery of the latent image without compromising processing speed.

References

- [1] Yuanchao Bai, Gene Cheung, Xianming Liu, and Wen Gao. Graph-based blind image deblurring from a single photograph. *IEEE Transactions on Image Processing*, 28:1404–1418, 2018. 2
- [2] Yuanchao Bai, Huizhu Jia, Ming Jiang, Xianming Liu, Xiaodong Xie, and Wen Gao. Single image blind deblurring

- using multi-scale latent structure prior. *IEEE Transactions on Circuits and Systems for Video Technology*, PP:1–1, 05 2019. [2](#)
- [3] T.F. Chan and Chiu-Kwong Wong. Total variation blind deconvolution. *IEEE Transactions on Image Processing*, 7(3):370–375, 1998. [1](#)
- [4] Liang Chen, Faming Fang, Tingting Wang, and Guixu Zhang. Blind image deblurring with local maximum gradient prior. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1742–1750, 2019. [1](#), [2](#), [6](#), [7](#), [8](#)
- [5] Liang Chen, Jiawei Zhang, Jinshan Pan, Songnan Lin, Faming Fang, and Jimmy S. Ren. Learning a non-blind deblurring network for night blurry images. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10537–10545, 2021. [3](#)
- [6] Xiaogang Chen, Xiangjian He, Jie Yang, and Qiang Wu. An effective document image deblurring algorithm. In *CVPR 2011*, pages 369–376, 2011. [2](#)
- [7] Xiaole Chen, Ruifeng Yang, Chenxia Guo, Shuangchao Ge, Zhihong Wu, and Xibin Liu. Hyper-laplacian regularized non-local low-rank prior for blind image deblurring. *IEEE Access*, 8:136917–136929, 2020. [2](#)
- [8] Hojin Cho, Jue Wang, and Seungyong Lee. Text image deblurring using text-specific properties. In *European Conference on Computer Vision*, 2012. [2](#)
- [9] Sunghyun Cho and Seungyong Lee. Fast motion deblurring. *ACM Trans. Graph.*, 28, 12 2009. [2](#), [4](#), [6](#)
- [10] Robert Fergus, Barun Singh, Aaron Hertzmann, Sam Roweis, and William Freeman. Removing camera shake from a single photograph. *ACM Trans. Graph.*, 25:787–794, 07 2006. [1](#), [2](#), [6](#)
- [11] Ankit Gupta, Neel Joshi, C. Zitnick, Michael Cohen, and Brian Curless. Single image deblurring using motion density functions. volume 6311, pages 171–184, 09 2010. [2](#)
- [12] Michael Hirsch, Christian Schuler, Stefan Harmeling, and Bernhard Schölkopf. Fast removal of non-uniform camera shake. pages 463–470, 11 2011.
- [13] Zhe Hu, Sunghyun Cho, Jue Wang, and Ming-Hsuan Yang. Deblurring low-light images with light streaks. In *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pages 3382–3389, 2014. [2](#)
- [14] Zhe Hu, Li Xu, and Ming-Hsuan Yang. Joint depth estimation and camera shake removal from single blurry image. In *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pages 2893–2900, 2014. [2](#)
- [15] Gary B. Huang, Manu Ramesh, Tamara Berg, and Erik Learned-Miller. Labeled faces in the wild: A database for studying face recognition in unconstrained environments. Technical Report 07-49, University of Massachusetts, Amherst, October 2007. [3](#), [4](#)
- [16] Neel Joshi, Richard Szeliski, and David J. Kriegman. Psf estimation using sharp edge prediction. In *2008 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8, 2008. [2](#)
- [17] Diederik Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *International Conference on Learning Representations*, 12 2014. [3](#)
- [18] Rolf Köhler, Michael Hirsch, Betty J. Mohler, Bernhard Scholkopf, and Stefan Harmeling. Recording and playback of camera shake: Benchmarking blind deconvolution with a real-world database. In *European Conference on Computer Vision*, 2012. [2](#), [5](#), [6](#)
- [19] Dilip Krishnan and Rob Fergus. Fast image deconvolution using hyper-laplacian priors. pages 1033–1041, 01 2009. [5](#)
- [20] Dilip Krishnan, Terence Tay, and Rob Fergus. Blind deconvolution using a normalized sparsity measure. In *CVPR 2011*, pages 233–240, 2011. [1](#), [2](#), [6](#)
- [21] Wei-Sheng Lai, Jia-Bin Huang, Zhe Hu, Narendra Ahuja, and Ming-Hsuan Yang. A comparative study for single image blind deblurring. In *IEEE Conferene on Computer Vision and Pattern Recognition*, 2016. [6](#)
- [22] Anat Levin, Yair Weiss, Fredo Durand, and William T. Freeman. Understanding and evaluating blind deconvolution algorithms. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1964–1971, 2009. [2](#), [5](#), [6](#)
- [23] Anat Levin, Yair Weiss, Fredo Durand, and William T. Freeman. Efficient marginal likelihood optimization in blind deconvolution. In *CVPR 2011*, pages 2657–2664, 2011. [1](#), [2](#), [4](#)
- [24] Lerenhan Li, Jinshan Pan, Wei-Sheng Lai, Changxin Gao, Nong Sang, and Ming-Hsuan Yang. Learning a discriminative prior for blind image deblurring. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6616–6625, 2018. [2](#)
- [25] Jing Liu, Jieqing Tan, Lei He, Xianyu Ge, and Dandan Hu. Blind image deblurring via local maximum difference prior. *IEEE Access*, 8:219295–219307, 2020. [2](#)
- [26] Marina Ljubenović and Mário A. T. Figueiredo. Blind image deblurring using class-adapted image priors. In *2017 IEEE International Conference on Image Processing (ICIP)*, pages 490–494, 2017. [2](#)
- [27] Caisheng Mao, Risheng Liu, and Haojie Li. Adaptive optimization with nested prior navigation for blind image deblurring. In *2018 IEEE Fourth International Conference on Multimedia Big Data (BigMM)*, pages 1–7, 2018. [2](#)
- [28] Xintian Mao, Yiming Liu, Fengze Liu, Qingli Li, Wei Shen, and Yan Wang. Intriguing findings of frequency selection for image deblurring, 2022. [1](#), [3](#)
- [29] Tomer Michaeli and Michal Irani. Blind deblurring using internal patch recurrence. volume 8691, pages 783–798, 09 2014. [2](#)
- [30] Jinshan Pan, Zhe Hu, Zhixun Su, Hsin-Ying Lee, and Ming-Hsuan Yang. Soft-segmentation guided object motion deblurring. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 459–468, 2016. [2](#)
- [31] Jinshan Pan, Zhe Hu, Zhixun Su, and Ming-Hsuan Yang. Deblurring face images with exemplars. pages 47–62, 09 2014. [2](#)
- [32] Jinshan Pan, Zhe Hu, Zhixun Su, and Ming-Hsuan Yang. Deblurring text images via l0-regularized intensity and gradient prior. In *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pages 2901–2908, 2014. [1](#), [2](#), [3](#), [4](#), [5](#), [7](#), [8](#)

- [33] Jinshan Pan, Deqing Sun, Hanspeter Pfister, and Ming-Hsuan Yang. Blind image deblurring using dark channel prior. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1628–1636, 2016. 1, 2, 6, 7, 8
- [34] Liyuan Pan, Richard Hartley, Miaomiao Liu, and Yuchao Dai. Phase-only image based kernel estimation for single image blind deblurring. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6027–6036, 2019. 2, 6
- [35] Qi Shan, Jiaya Jia, and Aseem Agarwala. High-quality motion deblurring from a single image. *ACM Trans. Graph.*, 27, 08 2008. 1, 2, 5, 6
- [36] Amanpreet Singh, Guan Pang, Mandy Toh, Jing Huang, Wojciech Galuba, and Tal Hassner. Textocr: Towards large-scale end-to-end reasoning for arbitrary-shaped scene text, 2021. 3, 4
- [37] Jian Sun, Wenfei Cao, Zongben Xu, and J. Ponce. Learning a convolutional neural network for non-uniform motion blur removal. pages 769–777, 06 2015. 2
- [38] Libin Sun, Sunghyun Cho, Jue Wang, and James Hays. Edge-based blur kernel estimation using patch priors. In *IEEE International Conference on Computational Photography (ICCP)*, pages 1–8, 2013. 2, 3, 4, 6
- [39] Yilun Wang, Junfeng Yang, Wotao Yin, and Yin Zhang. A new alternating minimization algorithm for total variation image reconstruction. *SIAM J. Imaging Sciences*, 1:248–272, 01 2008. 3
- [40] Zhou Wang, A.C. Bovik, H.R. Sheikh, and E.P. Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4):600–612, 2004. 6
- [41] Chen Wei, Wenjing Wang, Wenhan Yang, and Jiaying Liu. Deep retinex decomposition for low-light enhancement. *CoRR*, abs/1808.04560, 2018. 3, 4
- [42] Fei Wen, Rendong Ying, Peilin Liu, and Trieu-Kien Truong. Blind image deblurring using patch-wise minimal pixels regularization. *ArXiv*, abs/1906.06642, 2019. 2, 6, 7, 8
- [43] Oliver Whyte, Josef Sivic, Andrew Zisserman, and J. Ponce. Non-uniform deblurring for shaken images. *International Journal of Computer Vision*, 98:168–186, 06 2012. 6
- [44] Li Xu and Jiaya Jia. Two-phase kernel estimation for robust motion deblurring. In *European Conference on Computer Vision*, 2010. 2
- [45] Li Xu and Jiaya Jia. Depth-aware motion deblurring. In *2012 IEEE International Conference on Computational Photography (ICCP)*, pages 1–8, 2012. 2
- [46] Li Xu, Cewu Lu, Yi Xu, and Jiaya Jia. Image smoothing via l0 gradient minimization. volume 30, page 1, 12 2011. 3
- [47] Li Xu, Xin Tao, and Jiaya Jia. Inverse kernels for fast spatial deconvolution. In *European Conference on Computer Vision*, 2014. 2
- [48] Li Xu, Shicheng Zheng, and Jiaya Jia. Unnatural l0 sparse representation for natural image deblurring. In *2013 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1107–1114, 2013. 1, 2, 3, 4, 6
- [49] Yanyang Yan, Wenqi Ren, Yuanfang Guo, Rui Wang, and Xiaochun Cao. Image deblurring via extreme channels prior. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6978–6986, 2017. 2
- [50] Hong Zhang, Yujie Wu, Lei Zhang, Zeyu Zhang, and Yawei Li. Image deblurring using tri-segment intensity prior. *Neurocomputing*, 398, 02 2020. 2
- [51] Ziyu Zhang, Liangliang Zheng, Yongjie Piao, Shuping Tao, Wei Xu, Tan Gao, and Xiaobin Wu. Blind remote sensing image deblurring using local binary pattern prior. *Remote Sensing*, 14:1276, 03 2022. 2
- [52] Shicheng Zheng, Li Xu, and Jiaya Jia. Forward motion deblurring. pages 1465–1472, 12 2013. 2