Deep Plug-and-play Nighttime Non-blind Deblurring with Saturated Pixel Handling Schemes

Hung-Yu Shu Yi-Hsien Lin Yi-Chang Lu
National Taiwan University
{r09943021, d06943006, yiclu}@ntu.edu.tw

Abstract

Due to the setting of shutter speeds, over-exposed blurry images can often be seen in nighttime photography. Although image deblurring is a classic problem in image restoration, state-of-the-art methods often fail in nighttime cases with saturated pixels. The primary reason is that those pixels are out of the sensor range and thus violate the assumption of the linear blur model. To address this issue, we propose a new nighttime non-blind deblurring algorithm with saturated pixel handling schemes, including a pixel stretching mask, an image segment mask, and a saturation awareness mechanism (SAM). Our algorithm achieves superior results by strategically adjusting mask configurations, making our method robust to various saturation levels. We formulate our task into two new optimization problems and introduce a unified framework based on the plug-and-play alternating direction method of multipliers (PnP-ADMM). We also evaluate our approach qualitatively and quantitatively to demonstrate its effectiveness. The results show that the proposed algorithm recovers sharp latent images with finer details and fewer artifacts than other state-of-the-art deblurring methods.

1. Introduction

Non-blind image deblurring aims to remove blur artifacts from given blurry images and blur kernels. Mathematically, the general blurred degradation can be modeled as follows:

$$B = I \otimes K + n, \tag{1}$$

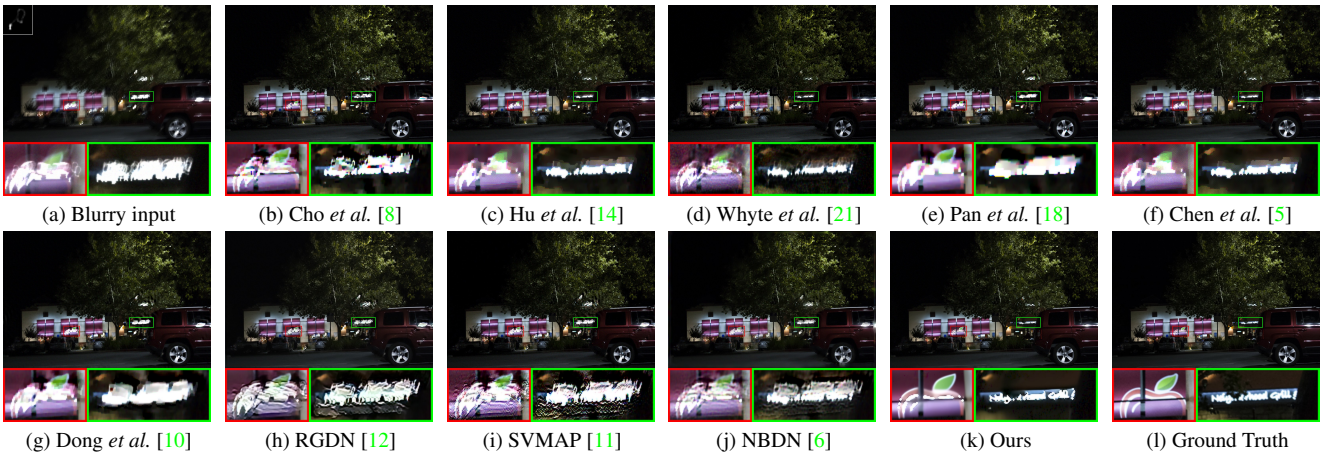
where B, K, I, n, and $\otimes$ denote the blurry image, blur kernel, sharp latent image, noise, and convolution operator, respectively. It is an ill-posed problem that can be estimated through the *maximum-a-posteriori* (MAP) approach. When solved by optimization-based methods, hand-crafted gradient sparsity priors, such as the hyper-Laplacian prior [15] or total variation priors [3, 19], are usually introduced as regularizers to ensure that the gradient of the recovered latent image is sparse.

Incorporating a deep prior learned from datasets in the MAP estimation problem has become popular in recent years. For example, Zhang *et al.* [25, 26] replaced the denoiser subproblem with a learned prior; Zhang *et al.* [23] unfolded every step of the iterative scheme as a neural network layer optimized via the backpropagation algorithm. Compared to gradient sparsity priors, deep priors retain finer image textures. Hence, a deep prior is also integrated into the proposed algorithm.

Camera sensors in nighttime photography frequently capture blurry and over-exposed images. Due to limited dynamic ranges of camera sensors, most existing algorithms fail in nighttime images since saturated pixels violate the general blurred degradation in Eq. (1). Hence, previous optimization-based methods [5, 8, 14, 18, 21] revised the general model to handle saturated pixels. Still, these methods often obtained over-smoothed deblurred results because of the usage of gradient sparsity priors. In the last few years, learning-based methods [6, 10–12] tried to learn the handling schemes and the data fidelity terms to correct the blur model from the statistics of images. However, if the model is applied to different saturation levels, the learned handling schemes can not well handle saturated pixels.

We propose a nighttime non-blind deblurring algorithm combining a deep learned prior and a set of saturated pixel handling schemes. The schemes include a pixel stretching mask, an image segment mask, and a saturation awareness mechanism (SAM). Compared to other work, our algorithm restores blurry nighttime images with finer details and better ringing suppression. The main contributions of this work are summarized as follows:

1) We design a unified PnP-ADMM deblurring framework with a deep-learned regularization term. The optimization flow can efficiently remove blurriness in nighttime images.
2) For highly-exposed images, we introduce an additional optimization stage to remove artifacts commonly seen in the non-saturated regions after deblurring. By blending this new result with the one obtained in the previous optimization stage, our approach can yield robust re-



(a) Blurry input (b) Cho *et al.* [8] (c) Hu *et al.* [14] (d) Whyte *et al.* [21] (e) Pan *et al.* [18] (f) Chen *et al.* [5]

(g) Dong *et al.* [10] (h) RGDN [12] (i) SVMAP [11] (j) NBDN [6] (k) Ours (l) Ground Truth

Fig. 1. Comparison of the deblurred results from the saturated dataset [14]. The results of robust optimization-based methods in $(b)-(f)$ generate fewer artifacts with specially designed schemes, but they cannot preserve fine structures. The results of learning-based methods in $(g)-(k)$ exhibit severe artifacts. On the contrary, the proposed method obtains a clear image with finer details and fewer artifacts.

sults in these challenging cases.

3) The experiment results on the synthetic datasets [6, 14, 17] and real blurry images demonstrate that our approach is comparable to the state-of-the-art methods in PSRN and SSIM values, and generates deblurred results with finer details and fewer artifacts visually.

2. Related Work

Due to the saturated pixels caused by long exposure time, the linear blur model often fails when applied to nighttime images, resulting in some severe ringing artifacts. Hence, several algorithms were proposed to solve this problem.

2.1. Optimization-based Methods

To better handle saturated pixels, Cho *et al.* [8] proposed an Expectation-Maximization (EM) framework that classified observed pixel intensities into inliers and outliers by discarding pixels that violate the assumption of the general blur model. As the analysis presented by Whyte *et al.* [21], the erroneous estimation of saturated pixels would generate ringing artifacts propagating throughout the image. Hence, Whyte *et al.* used an approximation function [4] to model saturated pixels and proposed a modified Richardson-Lucy (RL) framework that decouples saturated pixels from the whole image to reduce ringing. Hu *et al.* [14] integrated the methods of Cho *et al.* and Whyte *et al.* into an EM-based RL deconvolution algorithm to handle various types of outliers. They also decomposed an image into non-saturated and saturated regions to suppress ringing. Pan *et al.* [18] designed a hand-crafted function to mitigate the effect of outliers. However, the above methods needed heuristic parameter settings for the revised blur model. Therefore, Chen *et al.* [5] proposed a blur model without any heuristic settings to keep the blurry image within the range of camera sensors as follows:

$$\begin{aligned} & B = M \circ (I \otimes K) \\ \text{s.t.} \quad & M_i = \begin{cases} 1 & (I \otimes K)_i \leq 1 \\ \frac{1}{(I \otimes K)_i} & \text{Otherwise} \end{cases}, \end{aligned} \tag{2}$$

where i, I, K, M, and $\circ$ denote the pixel index, the sharp latent image, the blur kernel, the mask, and Hadamard product. M serves as a clipping function [21] to keep the blurry image within the range of camera sensors. Meanwhile, several studies [5, 7, 14, 18, 21] used the hyper-Laplacian prior [15] to address ill-posed issues. However, such methods often resulted in over-smooth recovered images and severe blocky effects due to gradient sparsity priors.

2.2. Learning-based Methods

In recent years, Zhang *et al.* [25, 26] designed a deep prior and integrated it into a MAP-based framework, and Gong *et al.* [12] proposed a universal gradient descent optimizer for image deconvolution. These methods preserved good image details but did not effectively handle saturated pixels, resulting in severe artifacts in saturation regions. Hence, recent work tried to train the data term to handle outliers during deconvolution. For instance, Dong *et al.* [10] proposed to learn the shrinkage function for the data term to handle severe outliers; Chen *et al.* [6] built a confidence estimation unit to correct the saturated pixels violating the blur model. In comparison, the model from Dong *et al.* [11] learned both spatially variant data and regularization terms in an end-to-end manner. However, learning-based saturated pixel handling schemes seemed not robust in datasets with different synthetic manners and even generated severe artifacts in over-exposed images.

3. Proposed Method

The robust optimization-based methods are good at handling saturated pixels, and the learning-based methods are great at retaining image textures during deconvolution. Hence, the proposed method leverages the merits of both types mentioned above. We propose a set of saturation handling schemes to alleviate the influence of saturated pixels. Besides, we use a deep learned prior as a regularization term to preserve more details during deconvolution. In the following, we elaborate on the proposed saturated handling schemes and then go through the PnP-ADMM framework.

3.1. Saturated Pixel Handling Schemes

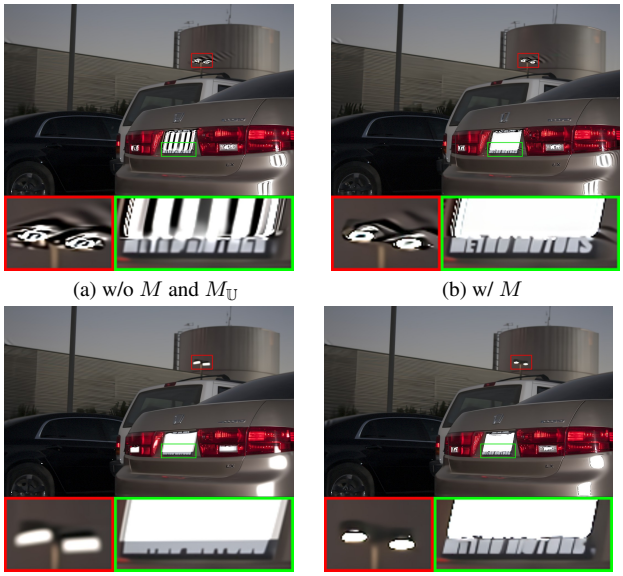
There are three parts to the proposed handling schemes. First, the pixel stretching mask M serves as a clipping function to make the value of the blurred image within the sensor range. Second, the image segment mask $M_{\mathbb{U}}$ excludes the saturated pixels to reduce ringing during deconvolution. Third, the saturation awareness mechanism decides when to enable the image segment mask $M_{\mathbb{U}}$.

Pixel Stretching Mask. The original linear blur model comes with severe artifacts in Fig. 2 (a), while the revised blur model using Eq. (2) better removes blur artifacts in Fig. 2 (b). Therefore, following the paradigm in [5], our basic blur model is formulated similarly to Eq. (2). In this work, we refer M as the pixel stretching mask generated through the function $\mathbf{\Psi}_M$, so that it can be distinguished from the two image segment masks introduced later. Each element in the mask $M = \mathbf{\Psi}_M(I, K)$ will be updated by the newly obtained I and K following Eq. (2) in each iteration.

Image Segment Mask. As suggested by [21], the ringing effect emerges from attempting to estimate the values of saturated pixels. Therefore, we propose to conduct a segmentation process in each iteration to decouple saturated pixels from the whole image. Based on the hard threshold ϕ, pixels are classified into two classes, a non-saturation region $\mathbb{U}$ and a saturation region $\mathbb{S}$. During deconvolution, the saturated pixels might also affect non-saturated pixels near the saturation region. Therefore, image dilation is adopted for the saturation region $\mathbb{S}$ to reduce impacts on boundaries. Different from [21], the proposed segmentation process considers the cross-channel consistency and the initial information from the observed blurred image B. For the cross-channel consistency, we use bitwise AND operations across masks of RGB channels to avoid color artifacts during deconvolution, and the segmentation process can be described as:

$$\begin{aligned} \mathbb{S} &= \bigcap_{c\in\{R,G,B\}} \{i \mid (I_i)_c \geq \phi\} \oplus D \oplus D_K, \\ \mathbb{U} &= \{i \mid I_i\} \setminus \mathbb{S}, \end{aligned} \tag{3}$$

where $(I_i)_c$, $c \in \{R, G, B\}$ denotes the separate RGB channels of the sharp latent image, $\oplus$ denotes the operator of image dilation, D denotes the structuring element of dilation, D_K denotes the structuring element of the blur kernel shape, i denotes pixel index and ϕ denotes the hard threshold.



(a) w/o M and $M_{\mathbb{U}}$ (b) w/ M (c) w/ $M_{\mathbb{U}}$ (d) w/ M and $M_{\mathbb{U}}$

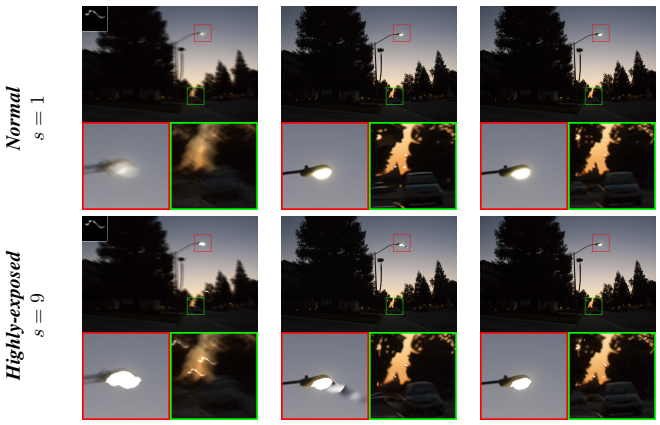
Fig. 2. Comparison results of different mask configurations. (a) Linear blur model without both masks. (b) Corrected blur model with M. (c) Separating saturated pixels during deconvolution with $M_{\mathbb{U}}$. (d) With both masks M and $M_{\mathbb{U}}$.

Then, we respectively define $M_{\mathbb{S}}$ and $M_{\mathbb{U}}$ as the binary masks corresponding to $\mathbb{S}$ and $\mathbb{U}$:

$$\begin{aligned} M_{\mathbb{S}} &= \mathbf{\Psi}_{M_{\mathbb{S}}}(I, K) = \begin{cases} 1 & \{i \mid I_i\} \in \mathbb{S} \\ 0 & \text{Otherwise} \end{cases}, \\ M_{\mathbb{U}} &= \mathbf{\Psi}_{M_{\mathbb{U}}}(I, K) = \begin{cases} 1 & \{i \mid I_i\} \in \mathbb{U} \\ 0 & \text{Otherwise} \end{cases}. \end{aligned} \tag{4}$$

Since the image segment mask $M_{\mathbb{S}}$ of the observed blurred image B encloses the largest possible saturation region, we also use it as the boundary mask $M_{\mathbb{S}}^{(0)} = \mathbf{\Psi}_{M_{\mathbb{S}}}(B, K)$ to reject sudden changes of pixel values in the non-saturation region. As shown in Fig. 2 (a) and (c), ringing near the saturation region can be reduced by separating saturated pixels during deconvolution. Through the masks M and $M_{\mathbb{U}}$, the deblurred result with fewer artifacts can be obtained in Fig. 2 (d).

Saturation Awareness Mechanism. Based on the observation of the degree of saturation, we propose the saturation awareness mechanism (SAM) for deblurring. Following the scheme proposed in [14] to synthesize the images, we demonstrate how the scale factor s, indicating the pixel range is changed from [0,1] to [0,s] before final clipping, can affect the observed blurred image B and the results of deblurring. As illustrated in blurry images of Fig. 3 (a), the larger the dynamic range before clipping is, the more obvious light



(a) B (b) w/o $M_{\mathbb{U}}$ (c) w/ $M_{\mathbb{U}}$

Fig. 3. Comparison of the deblurred results with different scale factors s. (a) Blurred image B, (b) Deblurred results I without the image segment mask $M_{\mathbb{U}}$, (c) Deblurred results I with the image segment mask $M_{\mathbb{U}}$.

streaks can be observed. Although the scale factor is not available for real-world images, we can borrow the concept and record the maximum pixel value $\tilde{s}$ during optimization:

$$\tilde{s} = max(I), \tag{5}$$

where I denotes the recovered latent image. The larger $\tilde{s}$ would indicate the image has a wider dynamic range, i.e. a higher degree of saturation. For images with large $\tilde{s}$, we call them *highly-exposed* images. As shown in the following paragraphs, we can apply the mask $M_{\mathbb{U}}$ to further remove artifacts in these challenging cases.

Fig. 3 (b) shows saturated pixels in the higher-scale-factor case come with more severe ringing artifacts. As shown in Fig. 3 (c), using the proposed segment mask $M_{\mathbb{U}}$, ringing can be suppressed in the red bounding box, and the same results of the non-saturation region can be preserved in the green one. The segment mask efficiently reduces ringing, but blur artifacts cannot be removed entirely due to the loss of information near saturated pixels. For the trade-off between ringing suppression and deblurring performance, the segment mask is activated for only *highly-exposed* images with severe ringing. For different degrees of saturation, the proposed SAM adopts proper mask configurations, making our method robust to different degrees of saturation.

3.2. Nighttime Non-blind Deblurring Algorithm

As shown in Fig. 3, we can suppress ringing artifacts by separating saturated pixels from the non-saturation region with the image segment mask $M_{\mathbb{U}}$. Hence, we formulate two optimization problems: one for image deblurring in both saturation and non-saturation regions, and the other for refining the non-saturation regions suffering severe ringing. Our variational model consists of two optimization problems in the proposed framework. If SAM identifies images that are *highly-exposed*, we blend the recovered latent images of two optimization problems. We summarize the proposed framework in Algorithm 1 and then define the energy function of two separable optimization problems.

Algorithm 1 Proposed Variational Model

```
for each iteration do
    solve for I using Equation (7)
    if highly-exposed then
        solve for I_U using Equation (8)
        I ← Blending(I, I_U)
    end if
end for
```

An optimization problem for deblurring both the saturated and non-saturated pixels:

$$\arg\min_{I} \mathcal{P}(M \circ (I \otimes K), B) + \lambda R(I), \tag{6}$$

where M is the pixel stretching mask in Eq. (2). The first term is a Poisson data term that enforces the similarity between (i) the Hadamard product of the pixel stretching mask and the convolution operation to the sharp latent image with the blur kernel and (ii) the blurred image, while the second term is a deep learned prior. To ease the computation burden, Eq. (6) can be modeled as follows:

$$\arg\min_{I} \mathcal{P}(I \otimes K, \frac{1}{M} \circ B) + \lambda R(I), \tag{7}$$

where each element of $\frac{1}{M}$ is a reciprocal of the corresponding element of M in Eq. (6). With this modification, a closed-form solution can be obtained by [22]. Since solving I within the Hadamard product in the original Eq. (6) requires a slower iterative approach, e.g., the conjugate-gradient method [7], the proposed model has advantages in computing time.

An optimization problem addressing the refinement of non-saturated pixels:

$$\arg\min_{I_{\mathbb{U}}} \mathcal{P}(M_{\mathbb{U}} \circ (I_{\mathbb{U}} \otimes K), M_{\mathbb{U}} \circ B) + \lambda R(I_{\mathbb{U}}) + \mathbb{I}_{\mathbb{R}_+}(I_{\mathbb{U}}), \tag{8}$$

where $M_{\mathbb{U}}$ is an image segment mask proposed in Eq. (4). The first is also a Poisson data term that focuses on the similarity of the non-saturation region between (i) the blurred image and (ii) the convolution result of the blur kernel and sharp latent image. The last two terms are a deep learned prior and the non-negativity constraint. Different from Eq. (7), we add a non-negative constraint because $\boldsymbol{M}_{\mathbb{U}}$ is a binary mask. That is to say, as $M_{\mathbb{U}}$ goes 0, the solution to $I_{\mathbb{U}}$ should stay positive. Therefore, we add a hard constraint to ensure the pixel value in $\mathbb{R}_+$. In the following paragraphs, we describe each term in the proposed model.

Poisson data term. Same with [5, 21], we assume the photon shot noise is modeled by the Poisson distribution and formulate the data fidelity term with a maximum likelihood estimator (MLE) as follows:

$$\begin{aligned}\mathcal{P}(k,\lambda) &= -log\sum_i \boldsymbol{Poisson}(k_i,\lambda_i) \\ &= 1^T\lambda - k^T\log(\lambda) + 1^T\log(k!),\end{aligned} \tag{9}$$

where k and λ are in the vectorized form.

Deep learned prior. Traditional gradient sparsity priors such as hyper-Laplacian [15] and total variation [3, 19] often lead to over-smooth results and even severe blocky effects. On the contrary, deep learned priors [25, 27] enforce desired natural image textures to mitigate ill-posed issues in the MAP estimation. Hence, we include the prior learned from the embedded statistics of images.

Indicator function. For the image quality, a non-negative constraint is incorporated to keep pixel values within the feasible domain as follows:

$$\mathbb{I}_{\mathbb{R}_+}(x) = \begin{cases}\inf, & x \notin \mathbb{R}_+ \\ 0, & x \in \mathbb{R}_+\end{cases}. \tag{10}$$

3.3. Numerical Algorithm with ADMM Solver

Next, we briefly summarize the solution-finding process of the proposed model. Both Eq. (7) and Eq. (8) are optimization problems with non-convex regularizers. Hence, we resort to ADMM algorithm [1] as the solver. The intuitive idea behind ADMM is to divide the problem into several subproblems, solving auxiliary variables of subproblems through proximal operators for different priors. $\boldsymbol{Prox}$ refers to the proximal operator [20], which is defined as:

$$\boldsymbol{Prox}_{F,\rho}(\boldsymbol{V}) = \arg\min_{\boldsymbol{Z}} F(\boldsymbol{Z}) + \frac{\rho}{2}||\boldsymbol{Z}-\boldsymbol{V}||_2^2, \tag{11}$$

where $F(\boldsymbol{Z})$ denotes the regularization term, and ρ denotes the penalty parameter of the auxiliary variable in the vectorized form. In the following paragraphs, we describe each proximal operator adopted in the proposed model, and derivation details can be found in the supplemental material.

Proximal operator for the Poisson data term. As demonstrated in [20], we can obtain the solution for the proximal operator for the MLE of the Poisson distribution as:

$$\boldsymbol{Prox}_{\mathcal{P},\rho}(\boldsymbol{V},\boldsymbol{B}) = -\left(\frac{1-\rho\boldsymbol{V}}{2\rho}\right) + \sqrt{\left(\frac{1-\rho\boldsymbol{V}}{2\rho}\right)^2 + \frac{\boldsymbol{B}}{\rho}}. \tag{12}$$

Proximal operator for the deep learned prior. Based on the plug-and-play trick [2, 26], we can take $\boldsymbol{V}$ as the noisy observation and $\boldsymbol{Z}$ as the noise-free image in Eq. (11), and the proximal operator can been seen as a Gaussian denoiser:

$$\boldsymbol{Prox}_{\lambda R,\rho}(\boldsymbol{V}) = \boldsymbol{Denoiser}(\boldsymbol{V}, \sqrt{\frac{\lambda}{\rho}}). \tag{13}$$

In this work, we use a pre-trained DRUNet [25], a specially designed CNN denoiser, for a plug-and-play framework. The physical meaning of $\sqrt{\frac{\lambda}{\rho}}$ is the denoiser strength, so we can adjust the denoiser strength based on the noise level in the observed blurred image.

Proximal operator for indicator function. The proximal operator of the indicator function can defined as:

$$\boldsymbol{Prox}_{\mathbb{I}_{\mathbb{R}_+}}(\boldsymbol{V}) = max(V_i, 0), \tag{14}$$

where $\boldsymbol{Prox}_{\mathbb{I}_{\mathbb{R}_+}}$ is a projection operator onto the convex set $\mathbb{R}_+$ and V_i denotes individual element in $\boldsymbol{V}$. With this proximal operator, we can enforce the non-negative constraint.

3.4. Optimizing Image I in Eq. (7)

Based on Eq. (9), we reformulate Eq. (7) as follows:

$$\arg\min_{\boldsymbol{I}} \mathbf{1}^T(\boldsymbol{KI}) - (\frac{\mathbf{1}}{\boldsymbol{M}} \circ \boldsymbol{B})^T log\,(\boldsymbol{KI}) + \lambda R(\boldsymbol{I}), \tag{15}$$

where $\boldsymbol{I}$, $\boldsymbol{B}$, $\boldsymbol{M}$ are in the vectorized form of I, B, M and $\boldsymbol{K}$ is the Toeplitz matrix of K. By following the method of ADMM [1] and introducing auxiliary variables $\boldsymbol{Z_1}, \boldsymbol{Z_2}$ and the multiplier $\boldsymbol{U}$, the solution to Eq. (15) can be seen as the solutions to the subproblems, $\boldsymbol{Z_1}$, $\boldsymbol{Z_2}$, $\boldsymbol{U}$ and $\boldsymbol{I}$. The solution to $\boldsymbol{Z_1}$-subproblem is the proximal operator for the Poisson data term in Eq. (12); the solution to $\boldsymbol{Z_2}$-subproblem is the pre-trained DRUNet denoiser (13); the solution to $\boldsymbol{U}$-subproblem is the update of multiplier; the solution to $\boldsymbol{I}$-subproblem is a closed-form solution which can be derived via Fourier transforms as well as element-wise multiplications and divisions with the circular boundary conditions [22]. After solving the above subproblems, we incorporate the mask update function $\boldsymbol{\Psi}_M$ to the ADMM flow. Derivation details and the summarized algorithm can be found in the supplemental material.

3.5. Optimizing Image $I_{\mathbb{U}}$ in Eq. (8)

Based on Eq. (9), we formulate Eq. (8) as:

$$\begin{aligned}\arg\min_{\boldsymbol{I}_{\mathbb{U}}} \mathbf{1}^T(\boldsymbol{M}_{\mathbb{U}} \circ \boldsymbol{KI}_{\mathbb{U}}) - (\boldsymbol{M}_{\mathbb{U}} \circ \boldsymbol{B})^T log\,(\boldsymbol{M}_{\mathbb{U}} \circ \boldsymbol{KI}_{\mathbb{U}}) \\ +\lambda R(\boldsymbol{I}_{\mathbb{U}}) + I_{\mathbb{R}_+}(\boldsymbol{I}_{\mathbb{U}}),\end{aligned} \tag{16}$$

where $\boldsymbol{I}_{\mathbb{U}}$, $\boldsymbol{B}$, $\boldsymbol{M}_{\mathbb{U}}$ are in the vectorized form of $I_{\mathbb{U}}, B, M_{\mathbb{U}}$ and $\boldsymbol{K}$ is the Toeplitz matrix of K. Besides $\boldsymbol{Z_2}$ used in Section 3.4, we introduce auxiliary variables $\boldsymbol{Z_3}$, $\boldsymbol{Z_4}$ and the multiplier $\boldsymbol{U}_{\mathbb{U}}$ using ADMM [1]. Then, we separately solve the subproblems of $\boldsymbol{Z_2}$, $\boldsymbol{Z_3}$, $\boldsymbol{Z_4}$, $\boldsymbol{U}_{\mathbb{U}}$ and $\boldsymbol{I}_{\mathbb{U}}$. The solution to $\boldsymbol{Z_2}$-subproblem is the pre-trained DRUNet denoiser (13); the solution to $\boldsymbol{Z_3}$-subproblem is the proximal operator of the Poisson data term in Eq. (12); the solution to $\boldsymbol{Z_4}$-subproblem is a projection operator in Eq. (14); $\boldsymbol{U}_{\mathbb{U}}$-subproblem is the update of multiplier. Because Toeplitz matrix $\boldsymbol{K}$ is coupled

with a Hadamard product with $\boldsymbol{M}_{\mathbb{U}}$, we use the conjugate-gradient method [7] to solve the $\boldsymbol{I}_{\mathbb{U}}$-subproblem instead of the fast closed-form method [22]. After solving these sub-problems, we integrate Eq. (4) of the mask update function $\boldsymbol{\Psi}_{M_{\mathbb{U}}}$ in ADMM flow. Derivation details and the summarized algorithm can be found in the supplemental material.

3.6. Overall Algorithm

After we separately solve the two optimization problems, we can unify the solutions to Eq. (7) and Eq. (8) as well as the SAM into an integrated framework shown in Algorithm 1. As shown in Fig. 3, highly saturated pixels come with more severe artifacts. Hence, the SAM decides whether to use the image segment mask to prevent artifacts from *highly-exposed* images based on two criteria: whether the maximum pixel value is larger than the hard threshold β and whether the time step t is in the first half of whole optimization process. Then, we blend $\boldsymbol{I}$ and $\boldsymbol{I}_{\mathbb{U}}$ with alpha blending as proposed in [21]. Besides, the regularization weight is adaptively adjusted based on the degree of saturation as follows:

$$\begin{aligned} \lambda^{(t+1)} &= \lambda^{(0)} \cdot \tilde{S}^{(t)} \\ \text{where} \quad \tilde{S}^{(t)} &= min(2^{\tilde{s}^{(t)}}, \alpha). \end{aligned} \tag{17}$$

The adaptive weight $\tilde{S}^{(t)}$ controls the regularization weight based on the maximum pixel value $\tilde{s}$ at the t^{th} iteration. The hard constraint α is designed to avoid over-large regularization weight. An early stopping mechanism is adopted to reduce the number of iterations by monitoring the relative residue of primal and dual variables. Implementation details can be found in the supplemental material.

4. Experiments

4.1. Parameter Setting and Runtime

In all experiments, the parameters are fixed to $\lambda = 2 \times 10^{-5}$, $\rho = 0.1$, $\alpha = 100$, $T = 100$, $\beta = 5.0$ and $\phi = 0.9$. The proposed algorithm is implemented in Pytorch 1.10.0. Executed on a computer with Intel Core i7-9700K CPU @ 3.60GHz and NVIDIA GeForce RTX 3080 Ti, obtaining a recovered image from a blurred image of $692 \times 1048 \times 3$ pixels takes 18.26 seconds if the high-exposed mode is not activated. With the *highly-exposed* mode, restoring an image with blur artifacts takes 20.26 seconds. Sensitivity analysis of hyper-parameters and run-time comparison are reported in the supplemental material.

4.2. Comparisons with State-of-the-Arts

In this section, we compare deblurred results with robust optimization-based methods [5, 8, 14, 18, 21] and state-of-the-art learning-based methods [6, 10–12, 25, 26]. We only use the non-blind deblurring steps of the blind deblurring algorithms [5, 14, 18]. For a fair comparison, all the experimental results are generated by publicly available code provided by the authors. To verify the robustness of all algorithms, we use three datasets [6, 14, 17] with different synthetic schemes, image scenes, and blur kernels as benchmarks. Besides, we also compare the deblurred results of real blurry images.

Table 1. Evaluations on the saturated dataset [14], low-illumination dataset [17] and nighttime dataset [6].

	Saturated [14]		Low [17]		Night [6]	
Methods	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
Cho [8]	27.95	0.9398	28.52	**0.9498**	29.43	0.9763
Hu* [14]	27.98	0.9380	25.30	0.9010	24.42	0.9246
Whyte [21]	26.13	0.8865	24.93	0.8908	24.86	0.9300
Pan* [18]	27.81	0.9471	27.00	0.9362	27.73	0.9647
Chen* [5]	27.57	0.9341	25.00	0.8968	24.70	0.9301
Dong [10]	25.39	0.9174	24.32	0.8981	26.31	0.9471
IRCNN [26]	20.89	0.8502	21.44	0.8508	25.32	0.9487
DPIR [25]	21.74	0.8618	21.83	0.8554	25.01	0.9392
RGDN [12]	24.36	0.8366	24.17	0.7293	30.23	0.9616
NBDN [6]	24.90	0.9237	22.82	0.8823	30.38	0.9815
SVMAP [11]	23.62	0.8800	25.45	0.9150	31.81	0.9810
Ours	**29.98**	**0.9548**	**28.58**	0.9366	**32.50**	**0.9832**

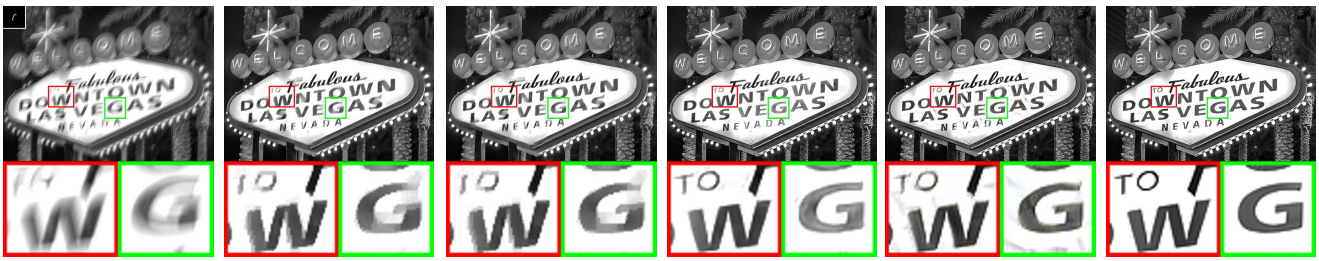
* :The non-blind deblurring step in the blind deblurring algorithm is used.

Saturated Dataset from Hu *et al.* [14] First, we use the dataset in [14] where noticeable light streaks of blurry images suggest the scenes are in the high degrees of saturation. As shown in Table 1, our method significantly outperforms the competing methods, improving 2dB in average PSNR. The deblurred results of this dataset are presented in Fig. 1. Optimization-based methods [5, 8, 14, 18, 21] generate fewer artifacts with specially designed saturated pixel handling schemes, but they often result in losses of image textures. For example, [8] and [18] have some false color artifacts around the saturation region. On the other hand, state-of-the-art learning-based methods [6, 10–12] can preserve more details but generate severe artifacts in both saturation and non-saturation region. Although [6] specially designed the confidence estimation unit and [11] was trained on images with saturated pixels, they still cannot handle saturated pixels with high-intensity values. That is to say, learning-based methods cannot well handle *highly-exposed* images so far. In comparison, our method not only preserves image details with the deep learned prior but also generates fewer artifacts due to the proposed saturated pixel handling schemes.

Low-illumination Dataset from Pan *et al.* [17] To evaluate the performance in low-illumination images, we test all algorithms using the dataset of [17] where the scale factor is 2.2. Our proposed algorithm achieves the competitive average PSNR values in Table 1, and the deblurred results are illustrated in Fig. 4. For instance, the optimization-based methods [8, 18] generate less ringing artifacts but more false color artifacts in the saturation region; the learning-based methods [10, 11] result in severe artifacts inside/near the saturation region. On the contrary, the proposed method generates fewer artifacts and finer details.

(a) Blurry input (b) Cho *et al.* [8] (c) Pan *et al.* [18] (d) Dong *et al.* [10] (e) SVMAP [11] (f) Ours

Fig. 4. Comparison of the deblurred results from the low-illumination dataset.



(a) Blurry input (b) Hu *et al.* [14] (c) Chen *et al.* [5] (d) NBDN [6] (e) SVMAP [11] (f) Ours

Fig. 5. Comparison of the deblurred results from the night dataset.

Night Dataset from Chen *et al.* [6] To further analyze the deblurring performance in night images, we also test all algorithms with the dataset in [6]. Unlike the dataset in [14, 17], the dataset contains diverse night scenes from skyscraper night views to shop signs with text patterns. Besides, the scale factor of the dataset is 1.2, and it is not as high as datasets in [14, 17]. The deblurred results are illustrated in Fig. 5 and our method outperforms all the methods evaluated in terms of PSNR and SSIM in Table 1. In this dataset, learning-based methods [10–12] perform better and preserve more details than the optimization-based methods via learned data and prior terms from embedded statistics of images. As shown in Fig. 5, deblurred results by [5, 14] exhibit common staircase patterns due to the gradient sparsity prior, and [11] come with undesirable ringing artifacts in the saturation region. Although [6] generates deblurred results without obvious artifacts, the proposed algorithm better restores concrete structures, *e.g.*, the text patterns in the green bounding box.

Real blurry images. For the analysis in real-world blurry images with saturation and noise, Fig. 6 shows deblurred results of a real blurry image from [16] with the kernel estimated by [18]. The methods [5, 6, 14, 26] cannot restore blurry artifacts in the saturation region. Besides, [6, 26] come with severe ringing in the saturation region, and the recovered images of [5, 14] suffer blocky effects. On the contrary, our method better restores blur artifacts with saturated pixels.

4.3. Analysis of Mask Configurations

Table 2 shows the quantitative results in the dataset of [6, 14, 17], and Fig. 2 demonstrates the qualitative results of mask configurations. Fig. 2 (a) shows severe artifacts if neither of the masks is adopted. Fig. 2 (b) suggests we can ameliorate artifacts inside the saturation region with M, and a better PSNR can be obtained. However, with only M, the SSIM of the saturated dataset [14] in Table 2 is even lower than the SSIM with no masks owing to the ringing from the obvious light streak. Meanwhile, Fig. 2 (c) reveals we can suppress artifacts near the saturation region using only $M_{\mathbb{U}}$. However, blur artifacts in/near the saturation region can not be thoroughly removed, degrading the performance in Table 2. Hence, M and $M_{\mathbb{U}}$ are adopted to remove ringing artifacts inside/near the saturation region for *highly-exposed* images. Fig. 2 (d) shows the best visual quality when using both masks.

Table 2. Comparison on the saturated dataset [14], low-illumination dataset [17] and nighttime dataset [6] w.r.t. different mask settings.

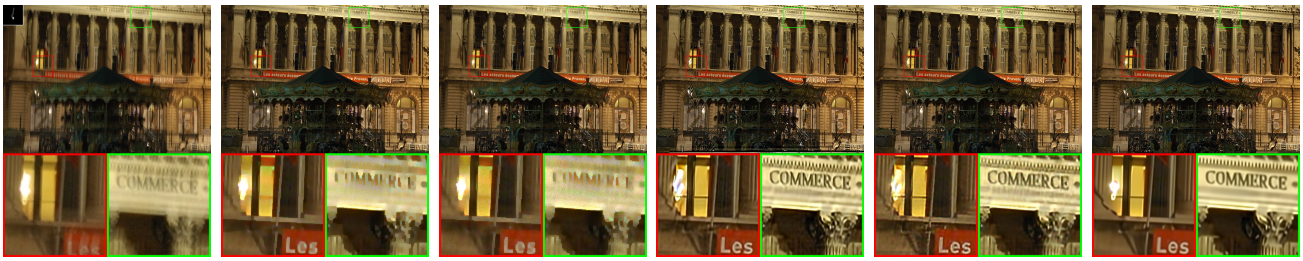
	Saturated [14]		Low [17]		Night [6]	
Mask settings	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
w/o both masks	26.51	0.9404	24.44	0.8883	31.31	0.9774
w/ M	27.55	0.9027	**28.58**	**0.9366**	**32.50**	**0.9832**
w/ $M_{\mathbb{U}}$	25.20	0.9269	24.25	0.9086	28.05	0.9683
w/ M and $M_{\mathbb{U}}$	**29.98**	**0.9548**	$^{-*}$	$^{-*}$	$^{-*}$	$^{-*}$

* denotes the *highly-exposed* mode is not enabled by SAM.

4.4. Analysis of Regularization Terms

In Section 3.2, we include a deep learned prior as a regularization term during deconvolution. Fig. 7 depicts the deblurred results using different regularization terms, *e.g.* hyper-Laplacian [15], total variation [3], guided filter [13], BM3D [9], SCUNet [24], FFDNet [27] and DRUNet [25].

Distinct from gradient sparsity priors [3, 15], DRUNet



(a) Blurry input (b) Hu *et al.* [14] (c) Chen *et al.* [5] (d) IRCNN [26] (e) NBDN [6] (f) Ours

Fig. 6. Deblurred results of real world blurred image with saturated pixels.

Table 3. Comparison on the saturated dataset [14], low-illumination dataset [17] and nighttime dataset [6] w.r.t. different regularizers.

	Saturated [14]		Low [17]		Night [6]	
Priors	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
HL† [15]	27.77	0.9076	28.49	0.9287	33.12	**0.9858**
TV† [3]	28.73	0.9310	**28.60**	0.9258	**33.32**	0.9838
BM3D [9]	29.38	0.9491	26.23	0.9130	25.19	0.9338
GF† [13]	28.91	0.9409	26.49	0.9151	25.44	0.9264
SCUNet [24]	21.85	0.7887	22.17	0.8440	28.48	0.9695
FFDNet [27]	23.25	0.8429	23.50	0.8734	28.54	0.9697
DRUNet [25]	**29.98**	**0.9548**	28.58	**0.9366**	32.50	0.9832

† HL:hyper-Laplacian, TV:Total Variation, GF:Guided Filter

[25] is good at preserving textures and has a less blocky effect. As shown in Fig. 7, the apple and the leaves can be retained with the pre-trained DRUNet. However, not all the learned priors are stable to saturated pixels. For example, the learned priors SCUNet [24] and FFDNet [27] generate color artifacts in the saturation region. BM3D [9] and the guided filter [13] retain sharper edges than gradient sparsity priors but generate artifacts in the saturation region. In addition to qualitative analyses, Table 3 shows the model with DRUNet as a regularizer comes with the competitive PSNR and SSIM in [6, 14, 17]. The performance of the proposed handling scheme is still competitive against state-of-the-art methods in Table. 1 even if the classical TV prior is used. Besides, although the PSNR and SSIM of gradient sparsity priors [3,15] outperform the DRUNet in the low-scale factor dataset [6], gradient sparsity priors with the large regularization weight have worse quantitative performance and the severe blocky effect in the high-scale factor dataset [14, 17] as shown in Fig. 7.

(a) Blurry input (b) hyper-Laplacian (c) Total Variation

(d) Guided Filter (e) BM3D (f) SCUNet

(g) FFDNet (h) DRUNet (i) Ground truth

Fig. 7. Comparison of deblurred results using different regularizers.

5. Conclusion

In this work, we propose a deep plug-and-play method for nighttime non-blind deblurring. Our handling schemes can effectively handle saturated pixels with different scale factors via a pixel stretching mask, an image segment mask, and a saturation awareness mechanism. Additionally, we develop a unified PnP-ADMM framework, together with the proposed handling scheme and the learned prior, to alleviate the impacts of saturated pixels and avoid the blocky effect efficiently. Finally, the comparisons with state-of-the-art methods demonstrate the superiority of our proposed algorithm on three benchmark datasets and real blurry images.

Acknowledgement

This work is partially supported by the National Science and Technology Council, Taiwan, under the grant numbers 112-2221-E-002-242; and by AUO Corporation, under the NTU project number 110HZT9B001. The authors would also like to thank National Center for High-performance Computing (NCHC), Taiwan, for providing computational and storage resources.

References

[1] Stephen Boyd, Neal Parikh, Eric Chu, Borja Peleato, Jonathan Eckstein, et al. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends® in Machine learning*, 3(1):1–122, 2011. 5

[2] Stanley H Chan, Xiran Wang, and Omar A Elgendy. Plug-and-play admm for image restoration: Fixed-point convergence and applications. *IEEE Transactions on Computational Imaging*, 3(1):84–98, 2016. 5

[3] Tony Chan, Selim Esedoglu, Frederick Park, A Yip, et al. Recent developments in total variation image restoration. *Mathematical Models of Computer Vision*, 17(2):17–31, 2005. 1, 5, 7, 8

[4] Chunhui Chen and Olvi L Mangasarian. A class of smoothing functions for nonlinear and mixed complementarity problems. *Computational Optimization and Applications*, 5(2):97–138, 1996. 2

[5] Liang Chen, Jiawei Zhang, Songnan Lin, Faming Fang, and Jimmy S Ren. Blind deblurring for saturated images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6308–6316, 2021. 1, 2, 3, 5, 6, 7, 8

[6] 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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10542–10550, 2021. 1, 2, 6, 7, 8

[7] Sunghyun Cho and Seungyong Lee. Fast motion deblurring. In *ACM SIGGRAPH Asia 2009 papers*, pages 1–8. 2009. 2, 4, 6

[8] Sunghyun Cho, Jue Wang, and Seungyong Lee. Handling outliers in non-blind image deconvolution. In *2011 International Conference on Computer Vision*, pages 495–502. IEEE, 2011. 1, 2, 6, 7

[9] Kostadin Dabov, Alessandro Foi, Vladimir Katkovnik, and Karen Egiazarian. Image denoising by sparse 3-d transform-domain collaborative filtering. *IEEE Transactions on image processing*, 16(8):2080–2095, 2007. 7, 8

[10] Jiangxin Dong, Jinshan Pan, Deqing Sun, Zhixun Su, and Ming-Hsuan Yang. Learning data terms for non-blind deblurring. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 748–763, 2018. 1, 2, 6, 7

[11] Jiangxin Dong, Stefan Roth, and Bernt Schiele. Learning spatially-variant map models for non-blind image deblurring. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4886–4895, 2021. 1, 2, 6, 7

[12] Dong Gong, Zhen Zhang, Qinfeng Shi, Anton van den Hengel, Chunhua Shen, and Yanning Zhang. Learning deep gradient descent optimization for image deconvolution. *IEEE Transactions on Neural Networks and Learning Systems*, 31(12):5468–5482, 2020. 1, 2, 6, 7

[13] Kaiming He, Jian Sun, and Xiaoou Tang. Guided image filtering. In *European conference on computer vision*, pages 1–14. Springer, 2010. 7, 8

[14] Zhe Hu, Sunghyun Cho, Jue Wang, and Ming-Hsuan Yang. Deblurring low-light images with light streaks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3382–3389, 2014. 1, 2, 3, 6, 7, 8

[15] Dilip Krishnan and Rob Fergus. Fast image deconvolution using hyper-laplacian priors. *Advances in neural information processing systems*, 22, 2009. 1, 2, 5, 7, 8

[16] Wei-Sheng Lai, Jia-Bin Huang, Zhe Hu, Narendra Ahuja, and Ming-Hsuan Yang. A comparative study for single image blind deblurring. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1701–1709, 2016. 7

[17] Jinshan Pan, Zhe Hu, Zhixun Su, and Ming-Hsuan Yang. l_0-regularized intensity and gradient prior for deblurring text images and beyond. *IEEE transactions on pattern analysis and machine intelligence*, 39(2):342–355, 2016. 2, 6, 7, 8

[18] Jinshan Pan, Zhouchen Lin, Zhixun Su, and Ming-Hsuan Yang. Robust kernel estimation with outliers handling for image deblurring. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2800–2808, 2016. 1, 2, 6, 7

[19] Yilun Wang, Junfeng Yang, Wotao Yin, and Yin Zhang. A new alternating minimization algorithm for total variation image reconstruction. *SIAM Journal on Imaging Sciences*, 1(3):248–272, 2008. 1, 5

[20] Gordon Wetzstein. Ee 367/cs 448i computational imaging and display notes: Noise, denoising, and image reconstruction with noise (lecture 10). 5

[21] Oliver Whyte, Josef Sivic, and Andrew Zisserman. Deblurring shaken and partially saturated images. *International journal of computer vision*, 110(2):185–201, 2014. 1, 2, 3, 5, 6

[22] Li Xu and Jiaya Jia. Two-phase kernel estimation for robust motion deblurring. In *European conference on computer vision*, pages 157–170. Springer, 2010. 4, 5, 6

[23] Jiawei Zhang, Jinshan Pan, Wei-Sheng Lai, Rynson WH Lau, and Ming-Hsuan Yang. Learning fully convolutional networks for iterative non-blind deconvolution. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3817–3825, 2017. 1

[24] Kai Zhang, Yawei Li, Jingyun Liang, Jiezhang Cao, Yulun Zhang, Hao Tang, Radu Timofte, and Luc Van Gool. Practical blind denoising via swin-conv-unet and data synthesis. *arXiv preprint arXiv:2203.13278*, 2022. 7, 8

[25] Kai Zhang, Yawei Li, Wangmeng Zuo, Lei Zhang, Luc Van Gool, and Radu Timofte. Plug-and-play image restoration with deep denoiser prior. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021. 1, 2, 5, 6, 7, 8

[26] Kai Zhang, Wangmeng Zuo, Shuhang Gu, and Lei Zhang. Learning deep cnn denoiser prior for image restoration. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3929–3938, 2017. 1, 2, 5, 6, 7, 8

[27] Kai Zhang, Wangmeng Zuo, and Lei Zhang. Ffdnet: Toward a fast and flexible solution for cnn-based image denoising. *IEEE Transactions on Image Processing*, 27(9):4608–4622, 2018. 5, 7, 8