Non-blind Deblurring Using Probabilistic Models and Spatial Adaptive Restoration

Chun-Lin Liao [1], Jian-Jiun Ding [2*], and Chun-Jen Shih [3]
[1] National Taiwan University, Taipei, Taiwan
E-mail: r11942060@ntu.edu.tw Tel: +886-2-33663662
[2] National Taiwan University, Taipei, Taiwan
E-mail: jjding@ntu.edu.tw Tel: +886-2-33669652
[3] National Taiwan University, Taipei, Taiwan
E-mail: f10942045@ntu.edu.tw Tel: +886-2-33663662

***Abstract*— The core concept of image deblurring is to establish a prior model and formulate the deblurring process as an optimization problem by using the statistical properties of gradients and noise. In usual, a user-defined parameter is introduced to control the trade-off between the sharpness of the deblurring result and the robustness to noise. However, using a single variable is difficult to make the deblurring algorithm adapt to different types of images. In this work, we aim to overcome this limitation. The criteria to determine different regions can be obtained through noise estimating and construct a space-variant noise model. Utilizing different statistical characteristics of noise, we can approximate their distribution using various hyper-Laplacian models. Thus, we establish two lookup mechanisms. The final result interpolates the result of different regions to obtain a smoother outcome. Additionally, for the significantly degraded image with a low sign-to-noise ratio (SNR), we apply an adaptive denoiser in the frequency domain to stabilize the deblurred image.**

I. INTRODUCTION

Image deblurring has been a thoroughly studied topic for the past several decades. Here, we focus on a specific type of distortion: motion blur [1,2,3] combined with additive white Gaussian noise (AWGN) [4]. This type of distortion is commonly observed in handheld photography, where the camera and the subject experience relative movement during the exposure time. The presence of AWGN, on the other hand, mainly stems from the sensor while reading discrete data, making the restoration process more challenging. In our study, we aim to address the challenges posed by motion blur and AWGN.

To begin with, the process of image blurring can be formulated as follows:

$$\mathbf{y} = \mathbf{x} \otimes \mathbf{k} + \mathbf{n}, \quad \mathbf{n} \sim \mathcal{N}(0,\ \sigma^2) \tag{1}$$

where $\mathbf{x}$ denotes sharp image, $\mathbf{k}$ is the point spread function (PSF), also known as the bur kernel, $\mathbf{y}$ is the image after motion blurring, $\mathbf{n}$ is the additive white Gaussian noise (AWGN) with standard deviation σ, and the notation $\otimes$ indicates 2D convolution in spatial domain. All kinds of uniform motion blurring can be simply modeled by (1). The goal is to reconstruct latent image $\mathbf{x}$ based on the blurry input $\mathbf{y}$ and $\mathbf{k}$.

II. RELATED WORKS

Non-blind deblurring algorithms can be roughly categorized into conventional methods and deep learning-based methods. Conventional methods, such as the Wiener filter, are derived based on minimum mean square error (MMSE), while conjugate methods are derived based on least squares error (LSE). Recently, the most popular framework is maximum a posteriori (MAP), which, through Bayes' theorem, maximizes the posterior probability $\mathrm{P}(\mathbf{x}|\mathbf{y},\mathbf{k})$ by maximizing the product of the likelihood $\mathrm{P}(\mathbf{y}|\mathbf{x},\mathbf{k})$ and the image prior $\mathrm{P}(\mathbf{x})$. A classic non-deep learning method utilizes a hyper-Laplacian function as the image prior [5], aiming to solve the following objective function

$$\min_{\mathbf{x}} \sum_{i=1}^{N} \left(\frac{\lambda}{2} (\mathbf{x} \otimes \mathbf{k} - \mathbf{y})_i^2 + \sum_{j=h,w} \left| (\mathbf{x} \otimes f_j)_i \right|^{\alpha} \right) \tag{2}$$

where N denotes the total number of pixels in the image, and f_h and f_w represent the vertical and horizontal gradient filters, respectively. The work suggests that the deblurring problem can be well modeled by the summation of a likelihood term and a hyper-Laplacian distribution prior $e^{-|\mathbf{x} \otimes f|^{\alpha}}$ with two adjustable parameters, α and λ.

To solve (2), half quadratic splitting (HQS) is introduced to separate the objective function into two sub-problems: deconvolution and image regularization. In the deconvolution phase, the latent image is obtained by direct FFT, while the image regularization phase is achieved through an analytic solution. The restored image is obtained by iteratively solving these two sub-problems.

Fig. 1. Deconvolution results from different λ fixing α.

Fig. 2. Deconvolution results from different α fixing λ.

Other non-DL methods follow a similar solving architecture. For example, BM3D-based deblurring [6, 7] utilizes the synthesis features of the BM3D framework, while outlier handling [8] removes saturated pixels during the deblurring process. With the advent of deep learning, methods have shifted. Instead of using the Hyper-Laplacian prior as in [7], the FCN method [12] suggests learning the image prior using a fully convolutional network. The IRCNN method [13], on the other hand, frames the image regularization problem as a denoising problem, solving it with a residual network. These methods are based on single optimization problems; however, image restoration should be adaptive to different images and various regions within a single image. Consequently, more complex models have been proposed. For instance, the NBDN [9] trains three separate models for restoration degree, image regularization, and a mask determining the probability of saturation. In INFWIND [10], deblurring is no longer treated as a single optimization problem; instead, it integrates Wiener filter deconvolution with image preprocessing and postprocessing.

III. Motivation

For non-DL methods, a single optimization problem restricts the degree of restoration in different regions of an image. As for the DL based methods, we find that for low SNR images, the learned priors are either ineffective or require the noise standard deviation to be known. Hence, blind denoising plays a crucial role in low SNR images deblurring. Our work is highly related to [5]. We first make a few observations about the hyper-Laplacian prior. In (2), the restoration depends on two parameters: λ and α . λ controls the sharpness of the deconvolution result. A greater λ will make the image prior term subtler, resulting in a sharper yet noisier image.

Additionally, α controls the curvature of the image prior and also affects the outcomes. A greater α indicates a more gradual decline in the gradient model, which corresponds to a sharper deconvolution.

We aim to replace the single hyper-Laplacian model with more accurate regularization terms according to local variance. The following section illustrates how we construct a spatially adaptive model, address the ringing effect, and provide the complete algorithm at the end.

IV. Proposed Method

A. Spatial Adaptive Probabilistic Model

In [5], the gradient is simply the difference between two horizontally and vertically adjacent pixels, i.e., $f_h = [1 \; -1]$, $f_w = [1 \; -1]^T$. Here, we use the Roberts cross filter [15] to calculate the gradient because the Roberts cross filter maximizes the KL divergence [14] of the gradient distribution between sharp images and blurry ones among conventional edge filters. Therefore, the filter in [5] is replaced by

$$f_h = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}, \; f_w = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}. \tag{3}$$

Generally speaking, when choosing λ, we consider two factors: the SNR of the blurry image and the local variance. For low SNR images, we want a smaller λ to increase the proportion of the image prior term. This supports the idea that the image prior term acts as a denoiser. In other words, denoising is more important than deblurring for low SNR images. On the other hand, local variance also determines the degree of restoration. For image patches with low variance, we can assume that the patch is almost constant. In these smooth patches, we apply strong regularization, forcing the gradients to be close to zero. We establish a 2D lookup table that uses noise standard deviation σ_n and local variance v_i to determine the desired λ. We use a rule of thumb to find the relationship between σ_n, v_i, and optimalλ_o. The experimental data consists of 80 undistorted real-world images. Each image is randomly blurred by one of the kernels in Levin [16] and AWGN with σ_n. We then deblur the blurry images using five different λ values: 0, 4, 200, 1000, and 10000. Note that when $\lambda = 0$, it means that only denoising is performed. Next, we calculate the local variance of the original image using a 9×9 block. For each patch $\mathbf{P}^i$ with variance v_i, we compare $\mathbf{P}^i$ with the patch in the same location in the five previously generated results using PSNR. By choosing the patch with the highest PSNR, we can find λ_o^i under σ_n and v_i.

Then, we partition the variance into 16 intervals with the following boundaries: 0, 1, 4, 8, 16, 30, 53, 88, 142, 212, 312, 449, 650, 961, 1467, and infinity. In each interval, there are several pairs of patches $\mathbf{P}_i$s with v_i falling within this range. For the parameters λ_o^is corresponding to these patches, we count the most frequent λ_o^i in this section and set it as the optimal lambda λ_o for the current variance section. To this end, we create a 1-D lookup table from local variance v_i to optimal λ_o under AWGN with noise standard deviation σ_n.

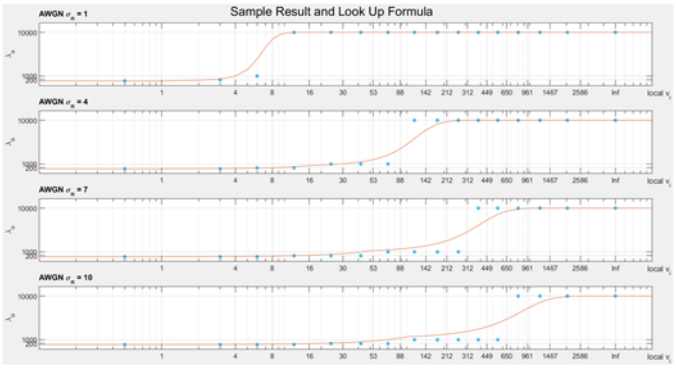

Fig. 3. Proposed 2D lookup table.

By repeating the above procedure for four different noise standard deviation, where $\sigma_n = 1, 4, 7,$ and 10, we obtain a scatter plot of λ_o versus σ_n and v_i, i.e., the blue dots in Fig. 3. Based on the experiment result, we use the following formula to model the scatter points

$$\lambda_{HighVar} = \frac{10000}{1+e^{\frac{-t}{\sqrt{\sigma_n^5}}}}, \quad t = v_i - (6.32 \cdot \sigma_n^2), \tag{4}$$

$$\lambda_o = \max\left(\lambda_{HighVar}(1-C), 0\right), \; C = \max\left(\frac{\sigma_n^2-(v_i-1)}{\sigma_n^2}, 0\right). \tag{5}$$

We use a sigmoid function for the high-variance intervals. The parameter C regulates the low-variance sections, ensuring that λ_o remains close to zero regardless of σ_n^2. Additionally, when v_i is less than 1, λ_o is set to 0, indicating that no deconvolution is needed. The 2D lookup table generated using (3) and (4) is shown in Fig. 3. by the red line.

So far, we have proposed a complete 2D lookup table for λ_o. To further improve the accuracy of our model, we adjust the hyper-Laplacian model [5] for different variance sections. We reintroduce the parameter b, which was neglected in [5], to achieve a more flexible model. Consequently, the objective function in (2) is rewritten as

$$\min_{\mathbf{x}} \sum_{i=1}^{N} \left(\frac{\lambda}{2} (\mathbf{x} \otimes \mathbf{k} - \mathbf{y})_i^2 + \sum_{j=h,w} \frac{\left|(\mathbf{x} \otimes f_j)_i\right|^{\alpha}}{b} \right) =$$

$$\min_{\mathbf{x}} \sum_{i=1}^{N} \left(\frac{\lambda b^{\alpha}}{2} (\mathbf{x} \otimes \mathbf{k} - \mathbf{y})_i^2 + \sum_{j=h,w} \left|(\mathbf{x} \otimes f_j)_i\right|^{\alpha} \right). \tag{6}$$

We sampled a total of 10,120 patches and classified them into the variance intervals mentioned above. For each variance section, we gather all the patches and used a pair of (α, b) to approximate the gradient distribution. Taking the local variance interval [312, 449] as an example, the optimal hyper-Laplacian prior is $\alpha = 0.9$ and $b = 11.9$.

B. Ringing Effect Suppression

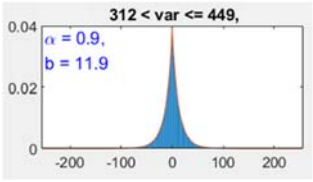

Fig. 4. Histogram of the gradient of loval variance between 312 and 449.

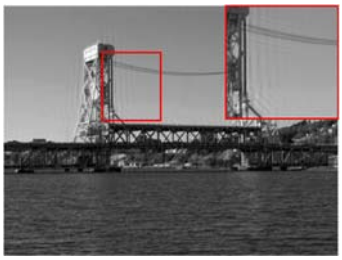

Fig. 5. Resulting image **I** from [5] that exhibits ringing effect.

Fig. 6. Lleft) Original image patch; (Right) After ringing effect removal.

An effective ringing effect method is to apply local subvoxel-shifts [17]. For every patch in the image, alternative sampling that minimizes the difference between adjacent pixels is applied. However, this method may have the problem of high computational complexity. Here, we propose an effective way of removing the ringing effect during the deconvolution phase.

Take the above image **I** for example, the black wires and electric tower against a bright background introduce sharp color transitions, which are therefore distorted by the ringing effect. However, we can observe that although the ripples are prominent, the gradients of these ripples are very close to zero. Utilizing this feature, we first calculate the gradient map of the entire image. For gradient values below a specific threshold (3 in our implementation), we smooth the ringing-distorted image across three scales: 100, 50, and 25. At the first scale of 100, we search for any 100 × 100 area where all gradients are zero. If such an area exists, we apply smoothing to the corresponding area in **I**. This procedure is then repeated for scales of 50 and 25. Next, we perform regional smoothing problem by solving the least-squares approximation using orthogonal projection matrices. Taking the scale of 100 as an example, for the top-left pixel in ringing patch, denoted as p_1, its x-index x_1 and y-index y_1 are set to 1. As we move from left to right, the x-index increases to 100, and as we move from top to bottom, the y-index also increases to 100. This approach provides a total of 10,000 3D data points, i.e., $[p_i, x_i, y_i]$. In other words, we aim to minimize

$$\|\mathbf{p} - (a_2\mathbf{v}_2 + a_1\mathbf{v}_1 + a_0\mathbf{v}_0)\|^2. \tag{7}$$

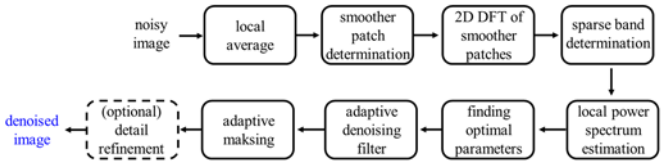

Fig. 7. Flowchart of the adaptive blind denoising algorithm

where $\mathbf{p}$, $\mathbf{v}_2$ and $\mathbf{v}_1$ corresponds to the vectorization of P_r, x-index and y-index, i.e.,

$$\mathbf{p} = \begin{bmatrix} p_1 \\ p_2 \\ \vdots \\ p_{10^4} \end{bmatrix}, \mathbf{v}_2 = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_{10^4} \end{bmatrix}, \mathbf{v}_1 = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_{10^4} \end{bmatrix}, \mathbf{v}_0 = \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix}. \quad (8)$$

Next, we concatenate $\mathbf{v}_0, \mathbf{v}_1, \mathbf{v}_2$ to form a matrix $\boldsymbol{C}=[\mathbf{v}_0, \mathbf{v}_1, \mathbf{v}_2]$. Following the orthogonal projection theorem, the coefficients of the estimated plane are given by

$$\begin{bmatrix} a_0 \\ a_1 \\ a_2 \end{bmatrix} = (\boldsymbol{C}^T\boldsymbol{C})^{-1}\boldsymbol{C}^T\mathbf{p}, \quad (9)$$

$$P_{\text{smooth}} = a_2 \begin{bmatrix} 1 & \cdots & 1 \\ \vdots & \ddots & \vdots \\ 100 & \cdots & 100 \end{bmatrix} + a_1 \begin{bmatrix} 1 & \cdots & 100 \\ \vdots & \ddots & \vdots \\ 1 & \cdots & 100 \end{bmatrix} + a_0. \quad (10)$$

C. *Blind Denoising*

Since denoising becomes crucial in deblurring low SNR images, we propose a blind denoising method that can restore images degraded by frequency-varying noise. The overall flowchart of the algorithm is shown here without delving into details.

The proposed method first estimates the noise spectrum, then performs local denoising using an adaptive Wiener filter. Our method can denoise a 512x512 grayscale image in under 0.5 seconds regardless of noise level, making it robust for deblurring, especially in low SNR images.

D. *Schematic of Implementation*

Finally, we provide a complete framework of the proposed non-blind deblurring algorithm along with a few implementation details. The variance map X_{pre} of the blurry image is calculated by the image with initial denoising and deblurring. We found that the correlation of variances between the blurry image and the sharp image is only 0.6. However, after performing denoising and deblurring, the correlation between the resulting image and the latent image can reach 0.9. Additionally, the output at each variance level is interpolated by a sigmoid-like function to prevent the image from having discontinuous transitions.

$$R = 1/(1 + \exp(-\frac{24(v-\frac{v_{i+1}+v_i}{2})}{v_{i+1}-v_i})) \quad (11)$$

Fig. 8. (Left) Deblurred result without denosing; (Right) Apply denoising.

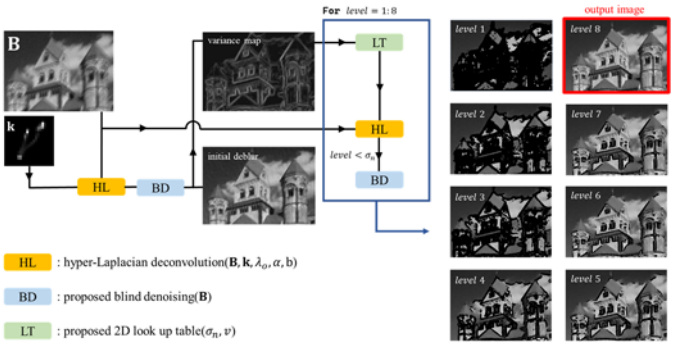

Fig. 9. Overall architecture of proposed non-blind deblurring algorithm.

Algorithm 1 Adaptive Deblurring Using 2D Look Up Table

Input: blurry image $\mathbf{B}$,
blur kernel $\mathbf{k}$,
noise std σ_n
Output: deblurred image S
$X_{pre} \leftarrow$ `proposed_denoise`$(\mathbf{B})$
$X_{var} \leftarrow$ variance(`fast_deconv`$(\mathbf{B}, 6000, 0.5)$)
for i = 1 : 8 **do**
  $X_{map} \leftarrow v_i \le X_{var} < v_{i+1}$
  $\lambda_o \leftarrow$ LookUp(v_i, σ_n)
  $X_l \leftarrow$(fast_deconv($\mathbf{B}$, λ_o, α_i, b_i))
  if i < σ_n **then**
    $X_l \leftarrow$ `proposed_denoise`(X_l)
    $X_l \leftarrow$ `proposed_ringing_removal`(X_l)
  end if
  $S \leftarrow S + X_{map} \circ (R \circ X_l + (1 - R) \circ X_{pre})$
  $X_{pre} \leftarrow X_l$
end for

Fig. 10. proposed non-blind deblurring algorithm

Also, we perform denoising and ringing effect removal only at levels where the variance is less than the noise standard deviation to avoid losing image details. In other words, if the image is not degraded by noise, we won't apply any denoising. Conversely, if the AWGN has a standard deviation greater than 8, we perform denoising at all levels. The complete process is outlined as in Fig. 10.

V. EXPERIMENTS

We examine our work by comparing the deblurring performance of motion blur with other methods under three different noise standard deviation. The test dataset comprises a total of five distinct image categories: text images, animal images [19, 20], night-time images [21], face images [18], and food images.

TABLE I. DEBLURRING PERFORMANCE UNDER $\sigma_n = 2$

PSNR	Blackboard (text)	Fox (Animal)	Boat (Night)	Man (Face)	Beefnoodle (Food)	**Average**
BM3D [6, 7]	**31.3842**	23.8940	21.9942	17.2226	20.0036	22.8997
Fast [5]	26.1323	34.0253	33.1821	30.1901	31.7080	31.0476
DMSP [11]	20.9645	28.7640	29.1063	25.4472	25.4122	25.9388
FCN [12]	27.1441	30.9976	30.8542	29.4330	29.6862	29.6230
IRCNN [13]	29.3836	33.0348	32.4839	29.7949	32.1332	31.3661
NBDN [9]	30.6898	34.6576	32.4464	29.7519	24.5466	30.4185
INFWIDE [10]	28.4646	34.0638	25.8255	28.3207	26.3456	28.6040
Proposed	30.8523	**34.9007**	**34.1105**	**31.7858**	**33.5533**	**33.0405**

TABLE II. DEBLURRING PERFORMANCE UNDER $\sigma_n = 5$

PSNR	Blackboard (text)	Fox (Animal)	Boat (Night)	Man (Face)	Beefnoodle (Food)	**Average**
BM3D [6, 7]	26.9748	27.9798	25.4035	20.9018	24.0873	25.0694
Fast [5]	25.3035	29.1439	28.5934	27.2970	28.0058	27.6687
DMSP [11]	20.8497	28.4093	28.7898	25.2838	25.3031	25.7271
FCN [12]	21.1616	26.8437	27.8374	24.5312	24.2157	24.9179
IRCNN [13]	27.0281	26.8644	25.9539	25.2703	25.7853	26.1804
NBDN [9]	**27.5403**	29.0457	27.1494	23.6711	19.5285	25.3870
INFWIDE [10]	27.2039	32.1745	27.4825	27.4527	26.1158	28.0859
Proposed	26.2209	**32.2095**	**32.1896**	**29.2618**	**31.2045**	**30.2173**

TABLE III. DEBLURRING PERFORMANCE UNDER $\sigma_n = 10$

PSNR	Blackboard (text)	Fox (Animal)	Boat (Night)	Man (Face)	Beefnoodle (Food)	**Average**
BM3D [6, 7]	23.8607	30.1757	27.9975	24.1482	26.5962	26.5557
Fast [5]	23.3875	24.1167	23.4251	23.0462	23.3501	23.4651
DMSP [11]	20.9026	**29.1435**	29.5945	26.0530	25.9833	26.3354
FCN [12]	21.2007	26.0540	27.3009	24.0345	23.6572	24.4495
IRCNN [13]	20.3963	16.9625	15.2924	15.9559	16.0320	16.9278
NBDN [9]	23.8940	21.7628	15.4941	18.1882	18.7295	19.6137
INFWIDE [10]	**25.8669**	27.3311	28.3032	25.9292	24.1002	26.3061
Proposed	25.4815	29.0070	**29.7646**	**27.5719**	**29.4297**	**28.2509**

We use the peak signal-to-noise ratio (PSNR) as our evaluation metric. For a sharp image $\mathbf{y}[m,n]$ and deblurred image $\mathbf{x}[m,n]$, the metrics is evaluated as follows:

$$\text{PSNR} = 10\log_{10}\frac{255^2}{\frac{1}{MN}\sum_{m=1}^{M}\sum_{n=1}^{N}|\mathbf{y}[m,n]-\mathbf{x}[m,n]|^2}. \quad (12)$$

From Tables I, II, and III, our method consistently achieves the best average performance among state-of-the-art techniques across three different scenarios.

Fig. 11. Effect of ringing artifact removal.

Fig. 12. Comparison of the details of the deblurring outputs.

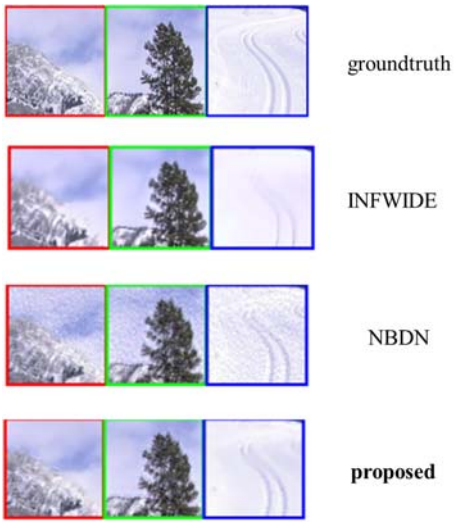

Fig. 13. Comparison of the details of the deblurring outputs.

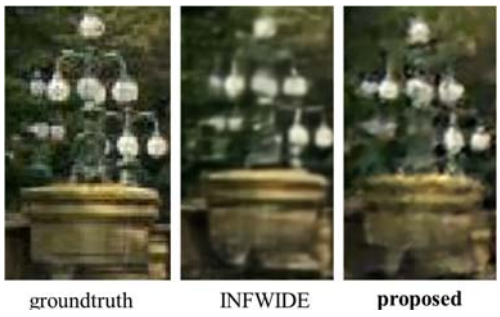

Fig. 14. Comparison of the detail of deblurring outputs.

At the first noise level ($\sigma_n = 2$), the results from these methods are similar. However, upon closer examination of image details, our method, which includes ringing effect removal, effectively eliminates the ripple on the roof that the DMSP does not address. Additionally, compared to the INFWIDE, our method better preserves texture details, such as the carvings on the wall.

At the second noise level, where AWGN begins to influence the results, shows that our method preserves tire tracks while also performing denoising. In contrast, the INFWIDE is more likely to sacrifice details for a more stable scene structure, and the NBDN does not specifically address denoising.

In high-noise situations ($\sigma_n = 10$), from Table III, the streetlight shows that our method tends to produce sharper edges, while the INFWIDE trades sharpness for overall image stability.

Moreover, in Figs. 11-14, the details of the deblurring outputs of the proposed method and other methods are compared. One can see that the deblurring outputs of the proposed method are much closer to the ground truths than those of other methods. With the proposed methods, the edges and texture of the object can be effectively reconstructed. Meanwhile, the proposed method can well avoid the ringing effect and the noise effect.

VI. Conclusion

In summary, we established a lookup table based on experimental data to map local variance to the hyper-Laplacian model and determine the optimal degree of deblurring. For low SNR images, we developed an efficient denoising method that handles noise varying in both space and frequency domains. By comparing our approach with state-of-the-art non-blind deblurring techniques, we demonstrate that our algorithm gives consistent performance across various types of degradation.

References

[1] H. Ji and C. Liu, "Motion blur identification from image gradients," in *IEEE/CVF Conf. Computer Vision and Pattern Recognition*, pp. 1-8, 2008, doi: 10.1109/CVPR.2008.4587537.

[2] G. Boracchi and A. Foi, "Modeling the performance of image restoration from motion blur," *IEEE Trans. Image Processing*, vol. 21, no. 8, pp. 3502-3517, 2012.

[3] Shengyang Dai and Ying Wu, "Motion from blur," in *IEEE/CVF Conf. Computer Vision and Pattern Recognition*, pp. 1-8, 2008, doi: 10.1109/CVPR.2008.4587582.

[4] C. Boncelet, "Image noise models," in *The Essential Guide to Image Processing*. Academic Press, Chap. 7, pp. 143-167, 2009.

[5] D. Krishnan and R. Fergus, "Fast Image Deconvolution using Hyper-Laplacian Priors," *Neural Information Processing Systems*, vol. 22, pp. 1-9, 2009.

[6] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, "Image denoising by sparse 3-D transform-domain collaborative filtering," *IEEE Trans. Image Processing*, vol. 16, no. 8, pp. 2080-2095, 2007.

[7] A. Danielyan, V. Katkovnik, and K. Egiazarian, "BM3D frames and variational image deblurring," *IEEE Trans. Image Processing*, vol. 21, no. 4, pp. 1715-1728, April 2012.

[8] S. Cho, J. Wang, and S. Lee, "Handling outliers in non-blind image deconvolution," in *Int. Conf. Computer Vision*, pp. 495-502, 2011.

[9] L. Chen, J. Zhang, J. Pan, S. Lin, F. Fang, and J. S. Ren, "Learning a non-blind deblurring network for night blurry images," in *IEEE/CVF Conf. Computer Vision and Pattern Recognition*, pp. 10542-10550, 2021.

[10] Z. Zhang, Y. Cheng, J. Suo, L. Bian, and Q. Dai, "INFWIDE: Image and feature space wiener deconvolution network for non-blind image deblurring in low-light conditions," *IEEE Trans. Image Processing*, vol. 32, pp. 1390–1402, 2023.

[11] S. A. Bigdeli, M. Zwicker, P. Favaro, and M. Jin, "Deep mean-shift priors for image restoration," *Advances in Neural Information Processing Systems*, vol. 30, pp. 1-7, 2017.

[12] J. Zhang, J. Pan, W. S. Lai, R. Lau, and M. H. Yang, "Learning fully convolutional networks for iterative non-blind deconvolution," in *IEEE Conf. Computer Vision and Pattern Recognition*, pp. 3817-3825, 2017.

[13] K. Zhang, W. Zuo, S. Gu, and L. Zhang, "Learning deep CNN denoiser prior for image restoration," in *IEEE Conf. Computer Vision and Pattern Recognition*, pp. 3929-3938, 2017.

[14] S. Kullback and R. A. Leibler, "On Information and sufficiency," *The Annals of Mathematical Statistics*, vol. 22, pp. 79–86, 1951.

[15] F. Nielsen, *Visual Computing: Geometry, Graphics, and Vision (Graphics Series)*, Charles River Media/Thomson Delmar Learning, 2005.

[16] A. Levin, Y. Weiss, F. Durand and W. T. Freeman, "Understanding and evaluating blind deconvolution algorithms," in *IEEE Conf. Computer Vision and Pattern Recognition*, pp. 1964-1971, 2009.

[17] E. Kellner, B. Dhital,V. G. Kiselev, and M. Reisert, "Gibbs-ringing artifact removal based on local subvoxel-shifts," *Magnetic Resonance in Medicine*, vol. 76, issue 5, pp. 1574-1581, 2016.

[18] T. Karras, S. Laine, and T. Aila, ""A style-based generator architecture for generative adversarial networks," in *IEEE/CVF Conf. Computer Vision and Pattern Recognition*, pp. 4401-4410, 2019.

[19] A. Kumar, "Bird species classification," Kaggle, 2019. [Online] Available: https://www.kaggle.com/datasets/akash2907/bird-species-classification.

[20] S. Banerjee, "Animal image dataset (90 different animals)," Kaggle, 2022. [Online] Available: https://www.kaggle.com/datasets/iamsouravbanerjee/animal-image-dataset-90-different-animals.

[21] A. Ye, "ExDark dataset," Kaggle, 2022. [Online] Available: https://www.kaggle.com/datasets/washingtongold/exdark-dataset.