This CVPR paper is the Open Access version, provided by the Computer Vision Foundation.
Except for this watermark, it is identical to the accepted version;
the final published version of the proceedings is available on IEEE Xplore.

Parameterized Blur Kernel Prior Learning for Local Motion Deblurring

Zhenxuan Fang[1] Fangfang Wu[1†] Tao Huang[1] Le Dong[1] Weisheng Dong[1†]
Xin Li[2] Guangming Shi[1]
[1]Xidian University [2]State University of New York at Albany
zxfang@stu.xidian.edu.cn wufangfang@xidian.edu.cn wsdong@mail.xidian.edu.cn

Abstract

Unlike global motion blur, Local Motion Deblurring (LMD) presents a more complex challenge, as it requires precise restoration of blurry regions while preserving the sharpness of the background. Existing LMD methods rely on manually annotated blur masks and often overlook the blur kernel's characteristics, which are crucial for accurate restoration. To address these limitations, we propose a novel parameterized motion kernel modeling approach that defines the motion blur kernel with three key parameters: length, angle, and curvature. We then use networks to estimate these kernel parameters, significantly improving the accuracy of blur kernel estimation. To effectively learn the motion blur representation, we incorporate a shared memory bank that stores blur prior information. Additionally, we introduce a dual-branch deblurring network: one branch leverages Mamba to capture long-range dependencies, while the other uses a mask-guided CNN focused on refining the local blurry regions. By fully utilizing the estimated bur prior information, our approach greatly enhances deblurring outcomes. Experimental results show that our method achieves state-of-the-art performance both quantitatively and visually, with a substantial reduction in computational complexity. The code is available at https://see.xidian.edu.cn/faculty/wsdong/Projects.

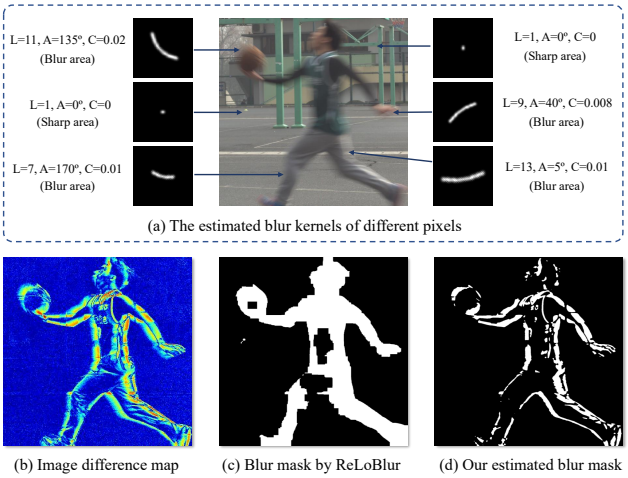
1. Introduction

Local Motion Deblurring (LMD) problem refers to removing blur caused by the motion of objects in specific regions. Unlike the global motion blur, where the entire image is uniformly blurred due to camera shake or overall scene motion, the blur kernels in LMD are typically spatially varying, as moving objects may exhibit different speeds, directions, and trajectories. Generally, a local blurry image can be modeled as the convolution of the sharp image with non-uniform blur kernels at each pixel, which can be expressed as:

$$\boldsymbol{B}(x,y) = \boldsymbol{\Omega}(x,y) \otimes \boldsymbol{K}(x,y) + \boldsymbol{N}(x,y), \qquad (1)$$

where $\boldsymbol{K}(x,y)$ represents the motion blur kernel at pixel coordinate (x,y), $\boldsymbol{\Omega}$ denotes the local neighborhood of the corresponding sharp image centered at pixel (x,y), $\boldsymbol{B}(x,y)$ is the observed blurry pixel, $\otimes$ denotes the convolution operation and $\boldsymbol{N}(x,y)$ is the additive Gaussian noise.

Previous approaches for solving the local deblurring problem are directly applying global deblurring methods [26, 33, 35]. However, these methods fail to distinguish between clear and blurry regions, leading to several issues: (1) Neglect of local blur characteristics: since local blur typically occurs in specific areas, applying a uniform deblurring process across the entire image will cause inadequate restoration of local details; (2) Introduction of artifacts: global deblurring methods often degrade sharp regions due



(a) The estimated blur kernels of different pixels

(b) Image difference map (c) Blur mask by ReLoBlur (d) Our estimated blur mask

Figure 1. (a) The estimated motion blur kernels in various regions have different parameters (**L**ength, **A**ngle and **C**urvature). (b) The difference map between the blur and GT image. (c) The blur mask by ReLoBlur [17] is coarse and inaccurate. (d) Our estimated fine-grained blur mask according to the length parameter of the motion kernels, which is close to the real blur difference map.

[†] Corresponding authors.

to deblurring the entire image, leading to the loss of original clarity and the introduction of artifacts [31]. Therefore, accurately estimating the blur kernels and masks is crucial for addressing the local motion deblurring problem.

Recently, various LMD methods have been proposed. In [17], LBAG method develops a local blur-aware gated network, including a blur detection module and several gate blocks. Researchers established the pioneering paired real local motion blur dataset known as ReLoBlur, which is captured by a synchronized beam-splitting photographing system. To supervise the blurry region detection, the ReLoBlur dataset includes the ground-truth local blur masks generated by a traditional foreground segmentation method [37], but these automatically generated masks contain errors, as seen in Figure 1 (c). Therefore, researchers in [18] manually annotated the local blur masks of the ReLoBlur dataset, which is very time-consuming and the masks remain relatively coarse, limiting the performance of high-quality restoration. Moreover, existing methods ignore the motion blur kernels of these blurry images, and the lack of information on the motion direction and amplitude of moving objects will weaken the deblurring results.

However, estimating the blur kernel is a highly ill-posed problem due to the lack of ground truth. To reduce estimation difficulty, we propose a parameterized approach to model the blur kernel, which represents the motion formation process by three essential parameters: length, angle, and curvature. Then, we can utilize the network to estimate these kernel parameters and construct the corresponding blur kernels. To improve the accuracy of blur kernel estimation, we integrate a memory bank in the parameter estimation network to store real-world motion prior information. An auxiliary blurring network is introduced to share this memory bank, ensuring consistent representation of the motion blur characteristics. Then we can derive the blur mask from the estimated kernel parameters. As shown in Figure 1 (a), the kernel length corresponds to the intensity of blur: a larger length indicates a greater degree of blur; a smaller length suggests that the pixel is nearly a sharp region. This effectively helps to distinguish between blurry and sharp areas at the pixel level, thereby obtaining a fine-grained blur mask. With the estimated blur kernel and mask, we propose a dual-branch deblurring network guided by the blur prior information: one branch utilizes the state space model (Mamba) to capture long-range dependencies and the other leverages a CNN to process local blurring.

Our method is motivated by three goals: (1) improve the accuracy of kernel estimation; (2) leverage blur prior information to enhance deblurring performance; (3) develop an effective and efficient local deblurring network. The main contributions are summarized as follows:

- We propose a novel motion kernel modeling method that utilizes interpretable parameters to characterize diverse blur kernels. This parameterized approach significantly improves kernel estimation accuracy and flexibility in real-world complex scenarios.
- We incorporate networks with a blur prior memory bank to estimate kernel parameters. The estimated parameters clearly indicate the blur degree of each pixel, which can be utilized to obtain fine-grained blur masks.
- Based on the estimated blur prior information, we propose a prior-guided deblurring network with two branches for both global and local processing. The U-shape Mamba branch captures long-range dependencies, while the CNN branch focuses on local blurry regions.
- Extensive experimental results demonstrate that the proposed method outperforms existing state-of-the-art methods both quantitatively and visually, with a significant advantage in computational complexity.

2. Related Work

2.1. Local Motion Deblurring

Global motion deblurring aims to restore the sharp image from the observed blurry image, which usually caused by overall scene motion. In contrast, local motion deblurring focuses on restoring specific areas that are blurred due to local motion objects. Researchers in [17] established the ReLoBlur dataset, which contains paired local motion blur and sharp images. They also proposed the Local Blur-Aware Gated network (LBAG), which introduces gating mechanisms to guide the network to focus on blurred areas. LMD-ViT [18] presented a vision transformer that utilizes adaptive window pruning transformer blocks, allowing unnecessary windows to be pruned based on the predicted confidence levels of blur regions. Recently, MUGNet [31] introduced a motion uncertainty-guided network that employs probabilistic representation models to address the complexities arising from motion uncertainty.

2.2. Motion Kernel Estimation

Motion kernels contain essential prior information about the motion trajectory of objects, which is critical for accurately restoring clear images in deblurring tasks. Earlier methods proposed various hand-crafted priors to constrain the solution space of blur kernels, utilizing techniques such as total variation [2] and ℓ_0-norm gradient prior [32]. Recently, many deep learning methods have made significant strides by explicitly estimating blur kernels. For instance, [25] predicts the probabilistic distribution of motion blur kernels at the patch level. In [22], blur kernels are estimated by performing division in Fourier space based on deep features. In UFPNet [6], the motion field is mapped to a latent space using normalizing flows, and the kernel estimation is implemented by predicting the corresponding latent code.

2.3. State Space Models

State Space Models (SSMs) [8, 9, 24] are sequence modeling methods from control theory that describe dynamic systems and are known for their excellent performance in capturing long-range dependencies. The Structured State-Space Sequence model (S4) [8] introduces parameter normalization through a diagonal structure. The Gated State Space layer [20] improves the S4 layer by incorporating additional gating units. S5 [24] proposes a MIMO SSM and simplifies state space layers for sequence modeling. Recently, Mamba [7] develops SSMs with a selective scan mechanism (S6) and outperforms Transformers in NLP while achieving linear computational complexity. Following this, numerous works have applied Mamba to vision tasks, including image segmentation [19, 28, 30], image classification [12, 36] and image restoration [10, 23].

3. Methodology

In this section, we present a novel approach for local motion deblurring. First, we propose a motion kernel modeling method, which utilizes interpretable parameters to characterize diverse blur kernels. Then we introduce a memory network to estimate kernel parameters and corresponding blur masks from blurry images. With these estimated motion priors, we design a prior-guided deblurring network with two branches to enhance deblurring performance.

3.1. Motion Kernel Modeling

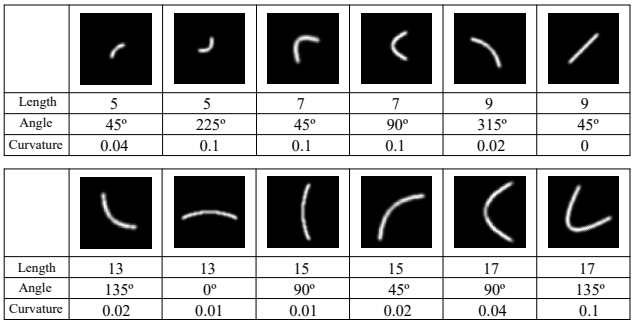
Traditional methods [1, 15] generate motion blur kernels by simulating random trajectories, which lack clear physical interpretability, making it difficult to intuitively understand motion properties such as direction and intensity. To address these limitations, we propose a parameterized motion kernel modeling method. We decompose the planar motion process into three essential parameters: $length$, $angle$ and $curvature$. Each parameter characterizes a distinct aspect of motion blur. By defining these parameters, we can mathematically calculate the motion trajectory and determine the pixel paths that constitute the blur kernel.

To generate a blur kernel $\boldsymbol{K} \in \mathbb{R}^{s\times s}$ of size s, we define the $center$ of kernel as the coordinate origin. Given the kernel parameters $length$ and $curvature$, the trajectory coordinate (x, y) in the kernel grid is computed as follows:

$$\begin{aligned} x &= center + t, \\ y &= center + curvature \cdot t^2, \end{aligned} \tag{2}$$

where t is a linearly spaced vector from $-(length//2)$ to $(length//2)$ that controls the horizontal displacement, and the $curvature$ affects the vertical displacement. Next, we convert the parameter $angle$ to $radians\ \theta$.

$$\theta = angle \times \frac{\pi}{180}. \tag{3}$$



Length	5	5	7	7	9	9
Angle	45°	225°	45°	90°	315°	45°
Curvature	0.04	0.1	0.1	0.1	0.02	0
Length	13	13	15	15	17	17
Angle	135°	0°	90°	45°	90°	135°
Curvature	0.02	0.01	0.01	0.02	0.04	0.1

Figure 2. Some examples of the motion kernels generated by different parameters, including Length, Angle and Curvature.

The rotation matrix $R(\theta)$ can be computed as

$$R(\theta) = \begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix}. \tag{4}$$

Then we can use the above rotation matrix to rotate the kernel, and the coordinates of the rotated point are given by:

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = R(\theta) \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}. \tag{5}$$

Finally, the blur kernel is normalized to ensure the sum of all pixels is equal to 1, which is expressed as:

$$\boldsymbol{K}_{\text{norm}} = \frac{\boldsymbol{K}}{\sum_{i=1}^{s}\sum_{j=1}^{s}\boldsymbol{K}(i,j)} \tag{6}$$

We use $\boldsymbol{K} = \boldsymbol{G}(length, angle, curvature)$ to represent the above generation process. Examples of the generated motion kernels with different parameters are shown in Figure 2. By introducing parameterized representation, the blur kernel generation process becomes transparent and easy to understand. Directly controlling these parameters greatly simplifies the generation process, allowing researchers to accurately design specific blur effects.

3.2. Motion Prior Estimation Network

Using the above motion blur kernel modeling method, the blur field of a blurry image can be represented by the motion parameters of each pixel. The blurry image formation process of Eq. (1) can be extended as follow

$$\boldsymbol{B}(x, y) = \boldsymbol{\Omega}(x, y) \otimes \boldsymbol{G}(\boldsymbol{P}(x, y)) + \boldsymbol{N}(x, y), \tag{7}$$

where $\boldsymbol{P}(x, y) = [length, angle, curvature]$ denotes the kernel parameters of pixel (x, y), $\boldsymbol{G}(\cdot)$ is the differentiable kernel generation process. This formulation allows for estimating kernel parameters through a reblurring framework.

We propose to use CNNs to estimate the kernel parameters of the blurry image. As illustrated in the upper of Figure

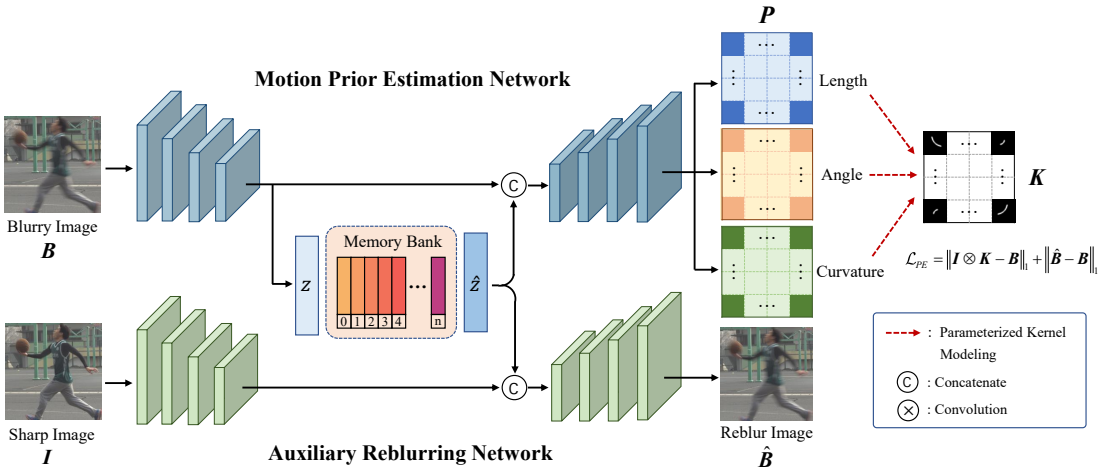


Figure 3. The architecture of the proposed motion prior estimation network (blue) that assembles with an auxiliary reblurring network (green), they share the same memory bank that stores rich blur-relevant prior information.

3, the blurry image $\boldsymbol{B} \in \mathbb{R}^{3\times H\times W}$ is first input into the encoder to extract image features $\boldsymbol{Z} \in \mathbb{R}^{c\times h\times w}$, which are used as the query to retrieve the most relevant items from the memory bank $\boldsymbol{M} \in \mathbb{R}^{n\times c}$, where n is the number of memory items. The memory bank is designed to store blur-relevant degradation features, making the updated features $\hat{\boldsymbol{Z}} \in \mathbb{R}^{c\times h\times w}$ to contain rich motion blur information.

The memory bank is updated by a self-supervised strategy [11]. We compute the cosine similarity between the i-th memory item $\boldsymbol{m}_i$ and the j-th vector $\boldsymbol{z}_j$ of $\boldsymbol{Z}$, expressed as

$$s_{ij} = \frac{\boldsymbol{m}_i \boldsymbol{z}_j^T}{\|\boldsymbol{m}_i\| \, \|\boldsymbol{z}_j\|}. \tag{8}$$

Then the index r of the most relevant memory item $\boldsymbol{m}_r$ for vector $\boldsymbol{z}_j$ is retrieved by $r = \arg\max_i s_{ij}$, and $\boldsymbol{m}_r$ is updated through $\boldsymbol{m}_r \leftarrow \eta \boldsymbol{m}_r + (1-\eta)\boldsymbol{z}_j$, where η is a decay rate. This process ensures the memory bank adapts to new input features while retaining motion-relevant information.

After updating the motion memory bank, we enhance the blur features $\boldsymbol{Z}$ by retrieving the memory items based on their similarity. The weight w_{ij} between each memory item $\boldsymbol{m}_i$ and the corresponding feature vector $\boldsymbol{z}_j$ is computed by a softmax operation $w_{ij} = \exp(s_{ij}) / \sum_{i=1}^{n} \exp(s_{ij})$. The motion-enhanced features are then obtained through linear aggregation of all memory items: $\hat{\boldsymbol{z}}_j = \sum_i^n w_{ij} \boldsymbol{m}_i$. The enhanced blur features $\hat{\boldsymbol{Z}}$ are concatenated with $\boldsymbol{Z}$ to incorporate additional motion degradation representations into the original features. The decoder subsequently predicts the kernel parameters $\boldsymbol{P} \in \mathbb{R}^{3\times H\times W}$, including the kernel length, angle and curvature. Then these parameters are employed to generate blur kernels by the proposed motion kernel modeling method in Sec. 3.1, which are used to perform spatially varying convolution on the sharp image. We aims to make the convolved image $\hat{\boldsymbol{B}}$ close to the observed blurry image $\boldsymbol{B}$, enabling accurate estimation of the kernel parameters from the input blurry image.

To strengthen the motion memory bank, we introduce an auxiliary reblurring network that utilizes the same memory bank to apply motion blur to sharp images, as illustrated at the bottom of Figure 3. The degradation features $\hat{\boldsymbol{Z}}$ not only assist the parameter estimation network in capturing motion prior information, but also empower the reblurring network to effectively insert complex motion blur patterns. This dual use of the memory bank facilitates more robust learning of motion representation, ultimately improving the accuracy of both parameter estimation and blur simulation processes. After the entire prior estimation network is trained, this auxiliary reblurring network will be discarded.

3.3. Prior-Guided Deblurring Network

To leverage the extracted prior information for enhacing image deblurring, we propose a **P**rior-**G**uided **D**eblurring **N**etwork (PGDN), which consists of a motion prior estimation network and a dual-branch deblurring network. The architecture of our PGDN is illustrated in Figure 4.

Motion Prior Estimation Network. In section 3.2, we have introduced the process of estimating motion kernel parameters from blurry images. With the estimated parameters, we can further derive the blur mask according to the length parameter. Specifically, the kernel length corresponds to the strength of the blur kernel: a larger length indicates a greater degree of blur, while a smaller length suggests that the pixel is nearly in a sharp region. As illustrated in Figure 4 (a), we have obtained the kernel parameters $\boldsymbol{P} = [\boldsymbol{P}_l, \boldsymbol{P}_a, \boldsymbol{P}_c]$ by the motion prior estimation network. For the kernel length $\boldsymbol{P}_l$, if it falls below threshold σ (i.e., $\boldsymbol{P}_l < \sigma$), the region is classified as sharp. Con-

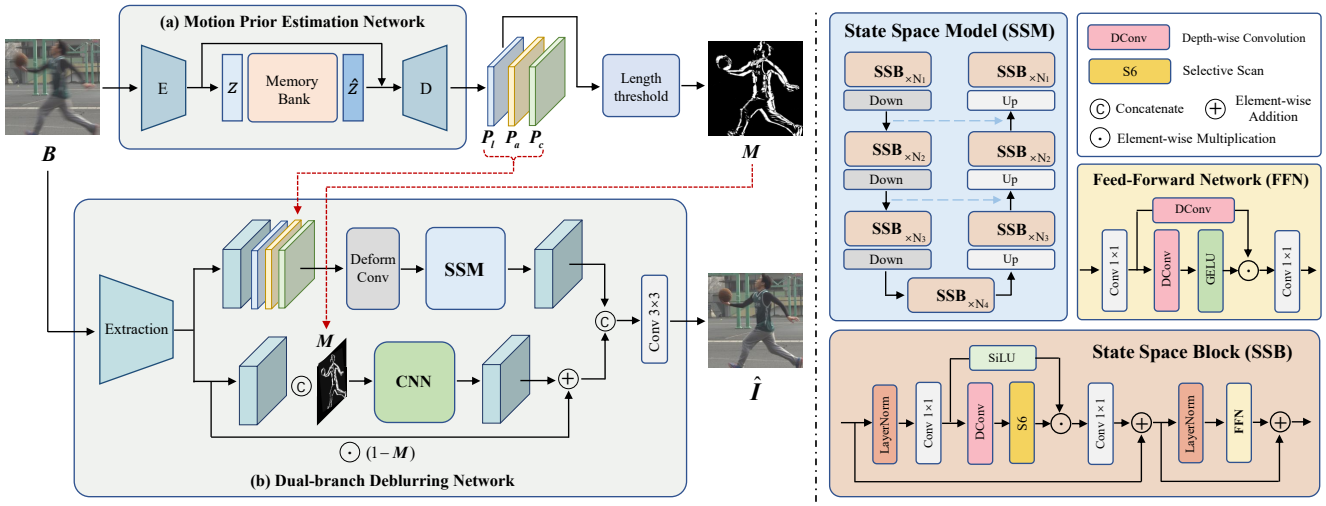


Figure 4. The architecture of the proposed PGDN, which includes a motion prior estimation network and a dual-branch deblurring network.

versely, if the length exceeds this threshold (i.e., $\boldsymbol{P}_l > \sigma$), the blur is considered significant, and the pixel is categorized as blurry region. This approach facilitates effective pixel-level distinction between blurry and sharp areas, enabling us to obtain a fine-grained binary blur mask $\boldsymbol{M}$.

Dual-branch Deblurring Network. To effectively leverage the estimated motion prior information, including blur kernel parameters and blur mask, we propose a dual-branch deblurring network that captures both global and local features. As shown in Figure 4 (b), the blurry image $\boldsymbol{B}$ is first processed by extraction networks to obtain shallow image features. In the global branch, these features are concatenated with the kernel parameters to incorporate motion blur information. Then the concatenated features pass through a deformable convolution layer, which dynamically adjusts the receptive field — a critical adaptation for handling spatially varying blur across different pixels. By learning the offsets and modulation, the deformable convolution enables the receptive field to adaptively adjust according to the blur kernel parameters.

After adjusting the receptive field of each pixel, we utilize state space models to capture long-range dependencies of the blur image features. Mathematically, the input signal $x(t)$ is mapped to the output $y(t)$ through implicit latent state $h(t)$, which can be formulated as the following linear ordinary differential equations (ODE):

$$h'(t) = \boldsymbol{C}h(t) + \boldsymbol{D}x(t), \quad y(t) = \boldsymbol{E}h(t) + \boldsymbol{F}x(t), \quad (9)$$

where $\boldsymbol{C}$, $\boldsymbol{D}$, $\boldsymbol{E}$ and $\boldsymbol{F}$ are learnable weight matrices. To apply this ODE in deep learning, the continuous signals are discretized by the zero-order hold technique, expressed as

$$\begin{aligned} &h_t = \bar{\boldsymbol{C}}h_{t-1} + \bar{\boldsymbol{D}}x_t, \quad y_t = \boldsymbol{E}h_t + \boldsymbol{F}x_t, \\ &\bar{\boldsymbol{C}} = e^{\Delta \boldsymbol{C}}, \quad \bar{\boldsymbol{D}} = (\Delta \boldsymbol{C})^{-1}\left(e^{\Delta \boldsymbol{C}} - I\right) \cdot \Delta \boldsymbol{D}, \end{aligned} \quad (10)$$

where Δ denotes the timescale parameter that convert the continuous parameters $\boldsymbol{C}$ and $\boldsymbol{D}$ into discrete parameters $\bar{\boldsymbol{C}}$ and $\bar{\boldsymbol{D}}$. Based on Eq. (10), Mamba [7] introduces a selective scanning mechanism (S6) with linear computational complexity. We propose a U-shape State Space Model (SSM) that consists of several State Space Blocks (SSB) with Downsample/Upsample layers at different scales. For more details about the selective scanning mechanism, please refer to the supplementary materials.

In the local branch, we integrate the image features with the estimated blur mask $\boldsymbol{M}$ to guide the network's attention toward the blurry areas. A lightweight CNN (e.g., several residual blocks) is then used to process these local features, as convolution operations have limited receptive fields and are less affected by irrelevant background pixels. Finally, we invert the blur mask and combine it with the enhanced image features to preserve the original sharp background.

3.4. Training Procedure

The training procedure of our PGDN includes two steps:

(1) Pre-train the motion prior estimation network. As illustrated in Figure 3, we apply the ℓ_1 loss to constrain the reblurred image close to the original blurry image $\boldsymbol{B}$:

$$\mathcal{L}_{PE} = \frac{1}{N}\sum_{i=1}^{N} \|\boldsymbol{I}_i \otimes \boldsymbol{K}_i - \boldsymbol{B}_i\|_1 + \|\mathcal{R}(\boldsymbol{I}_i) - \boldsymbol{B}_i\|_1, \quad (11)$$

where $\boldsymbol{I}_i$ denotes the sharp image, $\boldsymbol{K}_i$ is the motion blur kernels generated from the estimated parameters, and $\mathcal{R}(\cdot)$ represents the auxiliary reblurring network, N denotes the total number of training samples.

(2) Train the dual-branch deblurring network. Using the pre-trained estimation network, we utilize the extracted prior information to guide the deblurring network, the deblurred image is constrained to approximate the ground

Methods	ReLoBlur				ReLoBlur+GoPro				Params	FLOPs
	PSNR	SSIM	PSNR$_w$	SSIM$_w$	PSNR	SSIM	PSNR$_w$	SSIM$_w$	(M)	(T)
DeepDeblur [21]	33.05	0.8946	26.51	0.8152	33.02	0.8904	28.29	0.8398	11.72	17.13
DeblurGAN-v2 [16]	33.85	0.9027	27.37	0.8342	33.26	0.8975	28.29	0.8489	5.076	0.989
HINet [3]	34.36	0.9151	27.64	0.8510	34.40	0.9160	28.82	0.8672	88.67	8.696
MIMO-Unet [5]	34.52	0.9250	27.95	0.8650	34.64	0.9247	29.17	0.8766	16.11	7.850
Restormer [34]	34.76	0.9255	28.09	0.8661	34.85	0.9271	29.28	0.8785	26.13	6.741
Uformer-B [27]	35.03	0.9251	28.83	0.8719	35.14	0.9277	29.92	0.8865	50.88	4.375
NAFNet [4]	34.55	0.9260	27.97	0.8651	34.67	0.9269	29.21	0.8773	67.80	3.129
LBAG+ [17]	34.85	0.9257	28.32	0.8734	34.92	0.9318	29.30	0.8946	16.11	7.852
LMD-ViT [18]	N/A	N/A	N/A	N/A	35.42	0.9285	30.25	0.8938	54.50	1.485
PGDN (Ours)	**36.53**	**0.9340**	**29.15**	**0.8856**	**36.64**	**0.9351**	**30.87**	**0.8991**	29.21	0.613

Table 1. The comparison results on the benchmark testsets, the models are trained on the ReLoBlur dataset and ReLoBlur+GoPro dataset.

truth $\boldsymbol{I}$, expressed as :

$$\mathcal{L}_{deblur} = \frac{1}{N}\sum_{i=1}^{N} \|\boldsymbol{I}_i - \mathcal{F}(\boldsymbol{B}_i; \boldsymbol{P}_i; \boldsymbol{M}_i)\|_1, \qquad (12)$$

where $\mathcal{F}(\cdot)$ denotes the deblurring network, $\boldsymbol{P}_i$ and $\boldsymbol{M}_i$ are the kernel parameters and blur mask estimated from $\boldsymbol{B}_i$.

4. Experiments

4.1. Datasets and Implementation Details

Following previous local motion deblurring methods [17, 18], we adopt two training settings: (I) Train and test on the ReLoBlur [17] dataset, which contains 2010 image pairs for training and 395 pairs for testing. (II) Train on both the ReLoBlur and GoPro [21] datasets, test on the ReLoBlur testset. We use the Adam optimizer [13] ($\beta_1 = 0.9$ and $\beta_2 = 0.99$) to train the proposed PGDN for 800K iterations. The initial learning rate is 2×10^{-4} and is decreased when the training iteration reaches specific milestones. The training patch size is 256×256 and the batch size is 32.

4.2. Comparison with Existing Methods

Quantitative Results. We have compared our PGDN method with several state-of-the-art (SOTA) deblurring methods, including CNN-based methods ([3–5, 16, 17, 21]) and Transformer-based methods ([18, 27, 34]). As shown in Table 1, the models trained on ReLoBlur+GoPro datasets outperform those trained solely on ReLoBlur dataset, benefiting from a wider variety of scene samples and blur patterns. The proposed PGDN achieves superior results in terms of PSNR and SSIM metrics, significantly surpassing the SOTA method LMD-ViT over 1dB. To evaluate the restoration results in local blur areas, we also calculate the weighted PSNR and the weighted SSIM results following [17, 18]. Moreover, our method is not only high-performing but also more efficient, with fewer parameters and lower FLOPs, highlighting our effectiveness and efficiency.

Whyte [29]	UFPNet [6]	Our MPENet				PSNR	SSIM
		Baseline	KM	Memory	RN		
✓						36.69	0.9179
	✓					40.16	0.9631
		✓				39.56	0.9487
		✓	✓			41.37	0.9756
		✓	✓	✓		41.53	0.9759
		✓	✓	✓	✓	**41.76**	**0.9765**

Table 2. Ablation results between the original blurry image and the reblur image generated by different settings, which can reflect the accuracy of kernel estimation. KM denotes Kernel Modeling from estimated parameters, RN is the Reblurring Network.

Qualitative Results. The visual results of various methods are presented in Figure 5 and 6. Our method demonstrates a remarkable ability to restore sharper edges and details, especially in areas with complex textures and intricate patterns. In contrast, competing methods frequently leave residual blur or introduce artifacts. Notably, our approach effectively restores the blurry regions while preserving the clarity of the surrounding background.

4.3. Ablation Studies

Accuracy of Kernel Estimation. In this section, we assess the accuracy of the estimated blur kernels by the proposed Motion Prior Estimation Network (MPENet). Since motion blur kernels lack the corresponding ground truth, a common evaluation approach is calculating the fidelity between the original blurry image and the generated reblur image. We compare our method with several blur kernel estimation approaches, including the traditional non-uniform kernel estimation method by Whyte [29] and the normalizing flows-based method UFPNet [6]. As shown in Table 2, the accuracy of the baseline network (i.e., directly estimating kernels from network without employing kernel modeling) is higher than the traditional Whyte method, but lower than UFPNet. The proposed kernel modeling method significantly reduces the solution space of motion kernels,

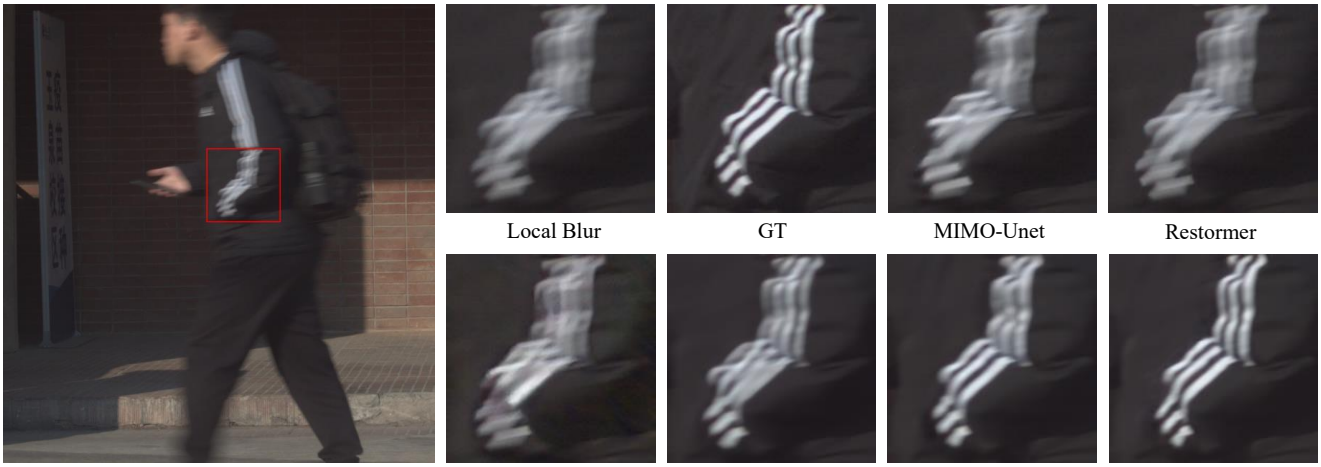


Figure 5. Visual comparisons on the ReLoBlur testset, the models are trained on the ReLoBlur dataset. From left to right: local blurry image, ground-truth, results by MIMO-Unet [5], Restormer [34], NAFNet [4], LBAG+ [17], LMD-ViT [18] and PGDN (ours).

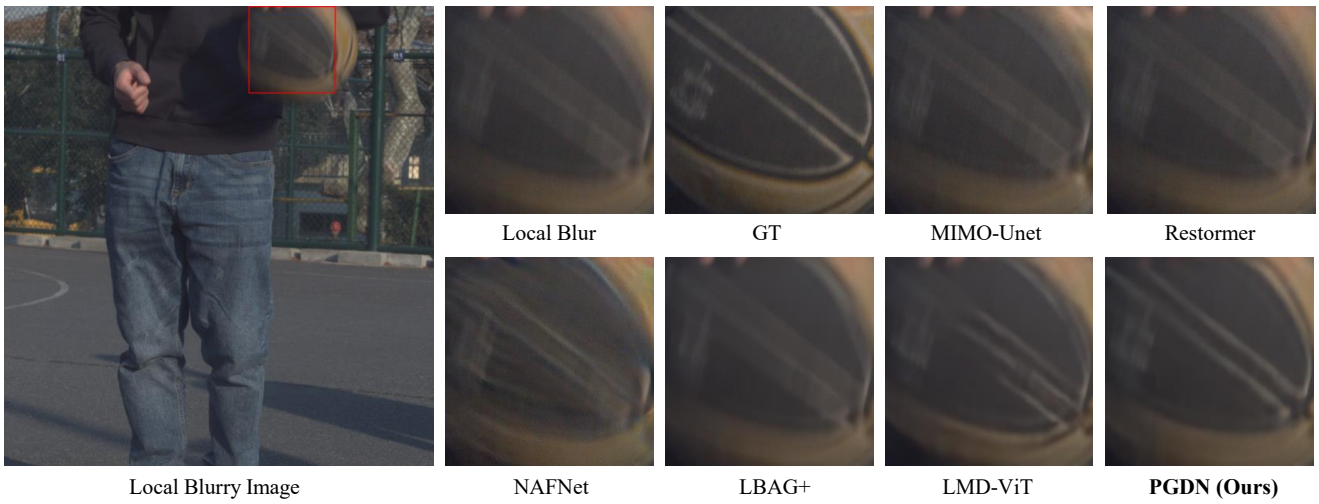
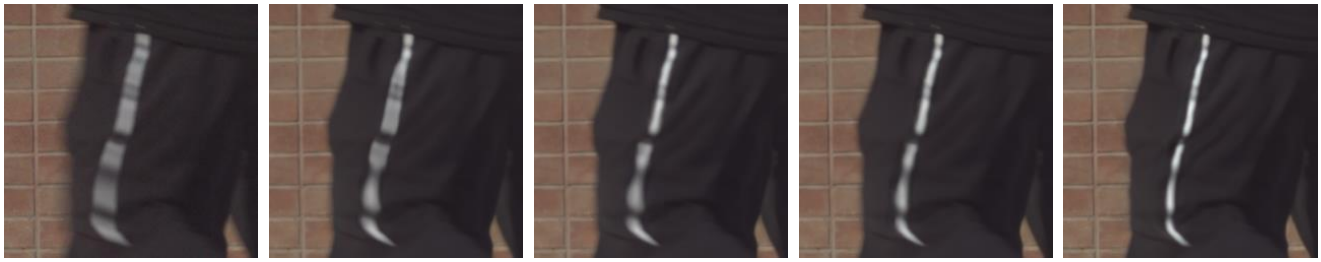


Figure 6. Visual comparisons on the ReLoBlur testset, the models are trained on the ReLoBlur+GoPro dataset.

thereby improving estimation accuracy. Additionally, the incorporation of the blur prior memory bank and the re-blurring network can further boost the accuracy of kernel estimation, thereby enhancing deblurring performance.

Effectiveness of Motion Prior. To demonstrate the effectiveness of the estimated motion prior information for improving deblurring performance, we conducted several ablation studies, including comparisons of directly using kernels versus utilizing kernel parameters, and the improvement by blur masks. In Table 3, the baseline model denotes our dual-branch architecture, which considers both global and local processing, serving as an effective baseline. It can be seen that incorporating kernel parameters produces better results than directly using kernels, because the kernel parameters already contain sufficient information about blur degradation, and the powerful representational capabilities of CNNs enable optimal utilization of this simplified information, leading to better deblurring performance. Furthermore, the guidance of blur mask allows the deblurring network to concentrate on restoring the blurry regions effectively, minimizing interference from irrelevant background details, as illustrated in the Figure 7.

Effectiveness of Network Modules. Comprehensive ablation studies are conducted to assess the effectiveness of various modules in the proposed dual-branch deblurring network. We replaced the State Space Model (SSM) with the CNN-based backbone NAFNet [4] and the Transformer-based Restormer [34]. As seen in Table 4, the SSM outperforms other alternatives while maintaining a lower computational cost, owing to its linear complexity and powerful



Local Blur | Baseline | Model-A | Model-B | **Ours**

Figure 7. Visual comparisons of different ablation models, our complete model can produce the sharpest edges.

Models	Kernel	Parameters	Mask	PSNR	SSIM
Baseline				35.47	0.9289
A	✓			35.83	0.9296
B		✓		36.09	0.9317
Ours		✓	✓	**36.53**	**0.9340**

Table 3. Ablation studies of different prior information for deblurring. Using kernel parameters produces better results than directly using kernel, and utilizing blur mask further boosts performance.

CNN	Transformer	SSM	PSNR	SSIM	Params (M)	FLOPs (T)
✓			34.97	0.9264	74.84	3.293
	✓		35.14	0.9271	33.17	6.905
		✓	**36.53**	**0.9340**	29.21	0.613

Table 4. Ablation studies of different backbone network, the SSM outperforms others while maintaining a lower computational cost.

Number of SSB	PSNR	SSIM	Params (M)	FLOPs (T)
[2, 2, 2, 2]	36.34	0.9291	21.82	0.463
[3, 3, 3, 3]	36.53	0.9340	29.21	0.613
[4, 4, 4, 4]	36.59	0.9354	36.60	0.762

Table 5. Ablation studies of the number of State Space Blocks.

abilities in capturing long-range dependencies. We also investigated the impact of the number of State Space Blocks (SSB) in Table 5. The results indicate that deblurring quality improves with an increasing number of blocks. To balance performance and computational complexity, we select a configuration of [3,3,3,3] for our network implementation. Furthermore, we validate the effectiveness of deformable convolution, which improves the results by 0.17 dB due to its ability to adjust the receptive field adaptively.

4.4. Extension to Global Motion Deblurring

To demonstrate the generalizability of our method in global deblurring, we conduct training and testing on the GoPro dataset [21]. As shown in Table 6, our approach achieves superior or comparable performance compared to existing global deblurring methods. This is due to our comprehensive utilization of motion degradation prior information.

Methods	GoPro		Params (M)	FLOPs (T)
	PSNR	SSIM		
DeepDeblur [21]	29.23	0.9160	11.72	17.13
DeblurGAN-v2 [16]	29.55	0.9340	5.076	0.989
MIMO-Unet [5]	32.45	0.9567	16.11	7.850
Restormer [34]	32.92	0.9611	26.13	6.741
Uformer-B [27]	33.06	0.9670	50.88	4.375
NAFNet [4]	33.71	0.9668	67.80	3.129
FFTformer [14]	34.21	0.9692	16.60	10.95
PGDN (Ours)	34.17	0.9501	29.21	0.613

Table 6. The results on the GoPro dataset. Our method can also achieve superior performance on global motion deblurring.

5. Conclusions

In this paper, we present a Prior-Guided Deblurring Network (PGDN) for addressing local motion deblurring. Our approach introduces a novel parameterized motion kernel modeling method, which can construct various motion blur kernels using three essential parameters: length, angle and curvature. We use a memory network to estimate these parameters within a reblurring framework, which significantly improves the accuracy of blur kernel estimation. Additionally, we propose a dual-branch deblurring network, where the global branch leverages Mamba to capture long-range pixel dependencies and the local branch employs a mask-guided CNN to concentrate on blurry areas. By fully utilizing the estimated blur prior information, our approach greatly improves deblurring results. Experimental results demonstrate that the proposed PGDN achieves state-of-the-art performance while also having significant computational efficiency. Future work will explore more methods for effectively integrating prior information into deblurring networks to improve performance further and extend this work from image to video by exploiting the temporal redundancy of local motion information.

Acknowledgments

This work was supported in part by the Natural Science Foundation of China under Grant 62472336, Grant 61991451 and Grant 62302369, and the Open Fund of Intelligent Control Laboratory 2024-ZKSYS-KF02-03. Zhenxuan Fang's work is supported by the Graduate Student Innovation Foundation of Xidian University under Grant YJSJ24012.

References

[1] Ayan Chakrabarti. A neural approach to blind motion deblurring. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part III 14*, pages 221–235. Springer, 2016. 3

[2] Tony F Chan and Chiu-Kwong Wong. Total variation blind deconvolution. *IEEE transactions on Image Processing*, 7 (3):370–375, 1998. 2

[3] Liangyu Chen, Xin Lu, Jie Zhang, Xiaojie Chu, and Chengpeng Chen. Hinet: Half instance normalization network for image restoration. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 182–192, 2021. 6

[4] Liangyu Chen, Xiaojie Chu, Xiangyu Zhang, and Jian Sun. Simple baselines for image restoration. In *European conference on computer vision*, pages 17–33. Springer, 2022. 6, 7, 8

[5] Sung-Jin Cho, Seo-Won Ji, Jun-Pyo Hong, Seung-Won Jung, and Sung-Jea Ko. Rethinking coarse-to-fine approach in single image deblurring. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 4641–4650, 2021. 6, 7, 8

[6] Zhenxuan Fang, Fangfang Wu, Weisheng Dong, Xin Li, Jinjian Wu, and Guangming Shi. Self-supervised non-uniform kernel estimation with flow-based motion prior for blind image deblurring. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 18105–18114, 2023. 2, 6

[7] Albert Gu and Tri Dao. Mamba: Linear-time sequence modeling with selective state spaces. *arXiv preprint arXiv:2312.00752*, 2023. 3, 5

[8] Albert Gu, Karan Goel, and Christopher Ré. Efficiently modeling long sequences with structured state spaces. *arXiv preprint arXiv:2111.00396*, 2021. 3

[9] Albert Gu, Isys Johnson, Karan Goel, Khaled Saab, Tri Dao, Atri Rudra, and Christopher Ré. Combining recurrent, convolutional, and continuous-time models with linear state space layers. *Advances in neural information processing systems*, 34:572–585, 2021. 3

[10] Hang Guo, Jinmin Li, Tao Dai, Zhihao Ouyang, Xudong Ren, and Shu-Tao Xia. Mambair: A simple baseline for image restoration with state-space model. *arXiv preprint arXiv:2402.15648*, 2024. 3

[11] Huaibo Huang, Aijing Yu, and Ran He. Memory oriented transfer learning for semi-supervised image deraining. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 7732–7741, 2021. 4

[12] Tao Huang, Xiaohuan Pei, Shan You, Fei Wang, Chen Qian, and Chang Xu. Localmamba: Visual state space model with windowed selective scan. *arXiv preprint arXiv:2403.09338*, 2024. 3

[13] Diederik P Kingma. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 6

[14] Lingshun Kong, Jiangxin Dong, Jianjun Ge, Mingqiang Li, and Jinshan Pan. Efficient frequency domain-based transformers for high-quality image deblurring. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5886–5895, 2023. 8

[15] Orest Kupyn, Volodymyr Budzan, Mykola Mykhailych, Dmytro Mishkin, and Jiří Matas. Deblurgan: Blind motion deblurring using conditional adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8183–8192, 2018. 3

[16] Orest Kupyn, Tetiana Martyniuk, Junru Wu, and Zhangyang Wang. Deblurgan-v2: Deblurring (orders-of-magnitude) faster and better. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 8878–8887, 2019. 6, 8

[17] Haoying Li, Ziran Zhang, Tingting Jiang, Peng Luo, Huajun Feng, and Zhihai Xu. Real-world deep local motion deblurring. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 1314–1322, 2023. 1, 2, 6, 7

[18] Haoying Li, Jixin Zhao, Shangchen Zhou, Huajun Feng, Chongyi Li, and Chen Change Loy. Adaptive window pruning for efficient local motion deblurring. In *The Twelfth International Conference on Learning Representations*, 2024. 2, 6, 7

[19] Jun Ma, Feifei Li, and Bo Wang. U-mamba: Enhancing long-range dependency for biomedical image segmentation. *arXiv preprint arXiv:2401.04722*, 2024. 3

[20] Harsh Mehta, Ankit Gupta, Ashok Cutkosky, and Behnam Neyshabur. Long range language modeling via gated state spaces. *arXiv preprint arXiv:2206.13947*, 2022. 3

[21] Seungjun Nah, Tae Hyun Kim, and Kyoung Mu Lee. Deep multi-scale convolutional neural network for dynamic scene deblurring. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3883–3891, 2017. 6, 8

[22] Christian J Schuler, Michael Hirsch, Stefan Harmeling, and Bernhard Schölkopf. Learning to deblur. *IEEE transactions on pattern analysis and machine intelligence*, 38(7):1439–1451, 2015. 2

[23] Yuan Shi, Bin Xia, Xiaoyu Jin, Xing Wang, Tianyu Zhao, Xin Xia, Xuefeng Xiao, and Wenming Yang. Vmambair: Visual state space model for image restoration. *arXiv preprint arXiv:2403.11423*, 2024. 3

[24] Jimmy TH Smith, Andrew Warrington, and Scott W Linderman. Simplified state space layers for sequence modeling. *arXiv preprint arXiv:2208.04933*, 2022. 3

[25] Jian Sun, Wenfei Cao, Zongben Xu, and Jean Ponce. Learning a convolutional neural network for non-uniform motion blur removal. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 769–777, 2015. 2

[26] Xin Tao, Hongyun Gao, Xiaoyong Shen, Jue Wang, and Jiaya Jia. Scale-recurrent network for deep image deblurring. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8174–8182, 2018. 1

[27] Zhendong Wang, Xiaodong Cun, Jianmin Bao, Wengang Zhou, Jianzhuang Liu, and Houqiang Li. Uformer: A general u-shaped transformer for image restoration. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 17683–17693, 2022. 6, 8

[28] Ziyang Wang, Jian-Qing Zheng, Yichi Zhang, Ge Cui, and Lei Li. Mamba-unet: Unet-like pure visual mamba for medical image segmentation. *arXiv preprint arXiv:2402.05079*, 2024. 3

[29] Oliver Whyte, Josef Sivic, Andrew Zisserman, and Jean Ponce. Non-uniform deblurring for shaken images. *International journal of computer vision*, 98:168–186, 2012. 6

[30] Renkai Wu, Yinghao Liu, Pengchen Liang, and Qing Chang. Ultralight vm-unet: Parallel vision mamba significantly reduces parameters for skin lesion segmentation. *arXiv preprint arXiv:2403.20035*, 2024. 3

[31] Zeyu Xiao, Zhihe Lu, Michael Bi Mi, Zhiwei Xiong, and Xinchao Wang. Unraveling motion uncertainty for local motion deblurring. In *ACM Multimedia 2024*. 2

[32] Li Xu, Shicheng Zheng, and Jiaya Jia. Unnatural l0 sparse representation for natural image deblurring. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1107–1114, 2013. 2

[33] Syed Waqas Zamir, Aditya Arora, Salman Khan, Munawar Hayat, Fahad Shahbaz Khan, Ming-Hsuan Yang, and Ling Shao. Multi-stage progressive image restoration. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 14821–14831, 2021. 1

[34] Syed Waqas Zamir, Aditya Arora, Salman Khan, Munawar Hayat, Fahad Shahbaz Khan, and Ming-Hsuan Yang. Restormer: Efficient transformer for high-resolution image restoration. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5728–5739, 2022. 6, 7, 8

[35] Hongguang Zhang, Yuchao Dai, Hongdong Li, and Piotr Koniusz. Deep stacked hierarchical multi-patch network for image deblurring. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5978–5986, 2019. 1

[36] Lianghui Zhu, Bencheng Liao, Qian Zhang, Xinlong Wang, Wenyu Liu, and Xinggang Wang. Vision mamba: Efficient visual representation learning with bidirectional state space model. *arXiv preprint arXiv:2401.09417*, 2024. 3

[37] Zoran Zivkovic. Improved adaptive gaussian mixture model for background subtraction. In *Proceedings of the 17th International Conference on Pattern Recognition, 2004. ICPR 2004.*, pages 28–31. IEEE, 2004. 2