Computers, Materials & Continua

Tech Science Press

Doi:10.32604/cmc.2025.059077

ARTICLE

Blur-Deblur Algorithm for Pressure-Sensitive Paint Image Based on Variable Attention Convolution

Ruizhe Yu[1], Tingrui Yue[3], Lei Liang[2,3,*] and Zhisheng Gao[1,*]

[1]School of Computer and Software Engineering, Xihua University, Chengdu, 610039, China
[2]College of Aerospace Engineering, Nanjing University of Aeronautics and Astronautics, Nanjing, 210016, China
[3]Low Speed Aerodynamics Institute, China Aerodynamics Research and Development Center, Mianyang, 621000, China

*Corresponding Authors: Lei Liang. Email: liangl94@nuaa.edu.cn; Zhisheng Gao. Email: gzs_xihua@xhu.edu.cn

Received: 27 September 2024; Accepted: 10 December 2024; Published: 06 March 2025

ABSTRACT: In the PSP (Pressure-Sensitive Paint), image deblurring is essential due to factors such as prolonged camera exposure times and high model velocities, which can lead to significant image blurring. Conventional deblurring methods applied to PSP images often suffer from limited accuracy and require extensive computational resources. To address these issues, this study proposes a deep learning-based approach tailored for PSP image deblurring. Considering that PSP applications primarily involve the accurate pressure measurements of complex geometries, the images captured under such conditions exhibit distinctive non-uniform motion blur, presenting challenges for standard deep learning models utilizing convolutional or attention-based techniques. In this paper, we introduce a novel deblurring architecture featuring multiple DAAM (Deformable Ack Attention Module). These modules provide enhanced flexibility for end-to-end deblurring, leveraging irregular convolution operations for efficient feature extraction while employing attention mechanisms interpreted as multiple 1 × 1 convolutions, subsequently reassembled to enhance performance. Furthermore, we incorporate a RSC (Residual Shortcut Convolution) module for initial feature processing, aimed at reducing redundant computations and improving the learning capacity for representative shallow features. To preserve critical spatial information during upsampling and downsampling, we replace conventional convolutions with wt (Haar wavelet downsampling) and dysample (Upsampling by Dynamic Sampling). This modification significantly enhances high-precision image reconstruction. By integrating these advanced modules within an encoder-decoder framework, we present the DFDNet (Deformable Fusion Deblurring Network) for image blur removal, providing robust technical support for subsequent PSP data analysis. Experimental evaluations on the FY dataset demonstrate the superior performance of our model, achieving competitive results on the GOPRO and HIDE datasets.

KEYWORDS: Pressure-sensitive paint; deep learning; image deblurring; typeset; variable attention convolution

1 Introduction

In the aerospace field, accurately measuring the surface pressure distribution of aircraft is crucial for enhancing performance, optimizing design, and ensuring safety. Recent advances in cooperative scientific research have enabled institutions to effectively address complex challenges by integrating expertise across multiple disciplines. Among these advancements, PSP (Pressure-Sensitive Paint) [1,2] has emerged as a promising technology for measuring pressure distributions on aircraft surfaces. PSP operates by reflecting pressure variations through changes in the optical properties of specially designed paint under varying pressure conditions. This non-contact measurement approach, enabled by interdisciplinary insights, has become widely adopted for aircraft pressure measurements, allowing researchers to address various aerodynamic design and operational challenges. Among current PSP methods, the light intensity-based pressure measurement technique has proven particularly mature and reliable. However, in practical applications, factors such as prolonged camera exposure times, camera shake [3], and model movement can introduce non-uniform blur into captured images, complicating subsequent pressure measurements and compromising data accuracy.

Image deblurring aims to restore clarity to images suffering from blur. Traditional non-deep learning deblurring methods are typically categorized into blind and non-blind techniques. Non-blind methods, which assume a known blur kernel, address deblurring as an inverse filtering problem and employ classical image deconvolution methods [4,5]. However, real-world blur conditions often require blind deblurring, in which both the blur kernel and the latent sharp image are estimated simultaneously, necessitating additional constraints to regularize the solution [6–8]. While these conventional methods perform well under specific conditions, they tend to fall short in real-world scenarios involving complex and ambiguous blur patterns.

With advancements in deep learning, significant progress has been made in addressing non-uniform motion blur. Techniques such as DeblurGAN and its improved version, DeblurGAN-v2, employ GANs (generative adversarial networks) to achieve high-quality deblurring without explicitly estimating a blur kernel [9,10]. Other approaches, including the scale-recurrent network and spatially variant deconvolution networks, utilize multi-scale and spatially adaptive techniques to handle dynamic scene deblurring more effectively [11,12]. Additionally, hierarchical models, such as deep stacked hierarchical multi-patch networks and spatially variant recurrent neural networks, have shown success in tackling complex motion blur through patch-based and recurrent processing frameworks [13,14]. These methods leverage deep network layers to generate deblurred outputs, circumventing the need for explicit blur kernel estimation that is often required by traditional techniques. Despite these advancements, several challenges persist. Many existing models rely on deep, stacked architectures and large image scales, resulting in exponential increases in computational complexity. Moreover, the uniform application of convolution across an entire image often leads to suboptimal deblurring performance in scenes with dynamically varying blur.

To address these challenges, this paper proposes a Variable Attention Convolutional Deblurring Network, optimized for the reduction of non-uniform motion blur. The proposed encoder-decoder-based architecture, named DFDNet (Deformable Fusion Deblurring Network), mitigates motion blur through stacked DAAM (Deformable Ack Attention Modules). The main contributions of this work are as follows:

1. This paper presents an encoder-decoder deblurring network, DFDNet, specifically designed for non-uniform motion blur. The architecture effectively captures spatial positional information from blurred images to produce sharp outputs.

2. The DAAM is introduced to capture motion information of varying amplitudes during processing. During the fusion attention phase, DAAM focuses on regions characterized by specific motion amplitudes, reducing the workload of each branch and generating variable convolution kernels tailored to objects of different sizes.

3. The wt and dysample operations are implemented during the upsampling and downsampling stages. Additionally, an RSCBlock is introduced during preprocessing to replace traditional convolutional upsampling, downsampling, and ResBlock operations. The wt and dysample serve as ultra-lightweight and efficient dynamic samplers that do not require high-resolution input features, significantly reducing parameter count. The RSCBlock enhances parameter efficiency and feature extraction capabilities by replacing traditional convolutions in ResBlocks with RSC convolution.

4. Experimental results demonstrate that the proposed method achieves effective deblurring on the FY dataset, as validated in subsequent pressure fitting analyses. Furthermore, DFDNet demonstrates competitive performance on the GOPRO [15] and HIDE [16] datasets.

2 Related Work

2.1 Deep Learning-Based Methods

Deep learning-based methodologies have made significant strides in addressing dynamic scene deblurring. Gao et al. [17] proposed a model employing parameter-selective sharing and nested skip connections, demonstrating effective restoration capabilities for complex motion blur scenarios. Cai et al. [18] introduced the Dark and Bright Channel Prior Embedded Network, which integrates specific priors to effectively tackle dynamic scene deblurring challenges. Suin et al. [19] presented a Spatially-Attentive Patch-Hierarchical Network that incorporates spatial attention mechanisms to adaptively manage motion deblurring tasks. Recently, Zamir et al. [20] developed a multi-stage progressive framework for image restoration, effectively restoring images through a series of iterative stages.

Further investigations have focused on enhancing architectural design and performance metrics. Mao et al. [21] proposed a Deep Residual Fourier Transformation approach that explores the use of frequency-domain transformations for mitigating blurriness. Subsequently, Mao et al. [22] extended their investigation into frequency selection strategies for achieving more effective deblurring outcomes. Zhou et al. [23] leveraged event data to facilitate low-light image deblurring, providing an efficient solution for challenging illumination conditions. Additionally, Lee et al. [24] introduced a Locally Adaptive Channel Attention-based Spatial-Spectral Neural Network aimed at optimizing deblurring performance.

2.2 Kernel Estimation Techniques

Kernel estimation methodologies for deblurring, which aim to estimate the blur kernel and recover the latent sharp image, have gained significant attention in recent years. Xu et al. [8] proposed a two-phase kernel estimation framework to effectively determine the motion blur kernel. Whyte et al. [25] addressed non-uniform image blur by utilizing a geometric model to estimate and mitigate the effects of camera shake. Harmeling et al. [26] introduced a space-variant single-image deconvolution algorithm, which proficiently mitigates camera shake through blind kernel estimation techniques. These kernel-based approaches emphasize the critical importance of precise kernel estimation in achieving successful image restoration.

2.3 Hybrid Techniques

Several recent works have integrated deep learning with conventional kernel estimation or other advanced concepts to achieve superior deblurring results. Li et al. [27] proposed a self-supervised deep expectation maximization approach for blind motion deblurring, providing a robust framework for complex motion blur.

A summary of all previously discussed methods is in Table 1. While these methods demonstrate a variety of effective image deblurring strategies, they often ignore the unique requirements associated with high-resolution PSP images. DFDNet addresses this gap by providing a dedicated architecture that incorporates a dual-network training framework specifically tailored to PSP images with variable convolution techniques. The proposed method ensures enhanced performance for restoring clarity from complex blurred images, which stands out from existing methods.

Table 1: A summary of the discussed methodologies

Methodology	Key feature	Limitation
Deep learning-based	Leveraging neural network architectures and priors for image deblurring	Struggle with high-resolution images like PSP
Kernel estimation	Estimation of blur kernels	Difficulty with complex motion and non-uniform blur
Hybrid techniques	Combination of kernel estimation and deep learning	Limited attention to specific high-resolution use cases

3 Method

DFDNet is a neural network specifically designed to address the problem of image deblurring for pressure-sensitive paint. The network is composed of two main components: a defuzzification module and a refuzzification module, which are jointly optimized using a consistency loss function. The primary goal of DFDNet is to recover a sharp image, denoted as I_s, from a blurred input image, I_b. Subsequently, a re-blurring network is introduced, which utilizes both I_s and I_b to generate a re-blurred image, I_r. During training, the convergence of both networks is carefully monitored. Upon achieving convergence of the network responsible for generating I_s, the intermediate output I_s in the re-blurring network is replaced with the deblurred output I_d. Following this replacement, the deblurring network undergoes further fine-tuning using a blur consistency loss to improve the quality of I_d. Notably, no normalization techniques are employed at any stage of this process.

3.1 De-Blur and Re-Blur Networks

In the shallow stage, two RSCBlock and wt downsampling are used to extract high-dimensional features, and two dysample upsampling are set at the end to recover the final spatial dimension. A convolutional layer is then employed to reduce the channel size of the feature map to 3 (RGB). Our findings indicate that simultaneously learning residual corrections, rather than directly learning the underlying sharp images, results in more stable and faster training. The blurred image, as shown in Eq. (1), contains all the signals from the sensor during the exposure time. Sharpening the image $I_{S(t)}$ at a specific time t* serves as the target, while the features extracted from I_b may correspond to an unexpected time t*. Where $g(\cdot)$ represents the camera response function (CRF), and T denotes the period of camera exposure. Instead of modeling the convolution kernel, we averaged M consecutive frames to produce a blurred image. Consequently, learning features from various times is easier than learning specific features directly.

In the middle of the deblurring network, we stacked six DAAM modules, as illustrated in Fig. 1. One of the RSC residual modules is depicted in Fig. 1a. The RSC module modifies the ResBlock residual module by replacing the standard convolution with SC convolution, which primarily exploits the spatial and channel redundancy between features for CNN compression, thereby reducing redundant calculations and promoting representative feature learning. The RSC module is mainly composed of the SRU (Spatial Reconstruction Unit) and the CRU (Channel Reconstruction Unit). Specifically, for the intermediate input feature X in the bottleneck residual block, we first obtain the spatial thinning feature X^w through spatial reconstruction. Subsequently, we derive the channel thinning feature Y using the channel reconstruction module. Finally, the SCConv module leverages the spatial and channel redundancy between features to reduce redundancy between feature graphs and enhance the CNN feature representation.

$$I_b = g\left(\frac{1}{T}\int_{t=0}^{T} I_{S(t)}dt\right), I_b \simeq g\left(\frac{1}{M}\sum_{t=0}^{M-1} I_{S[t]}\right) \tag{1}$$

The image re-blurring architecture, as depicted in Fig. 2, addresses the challenge of the excessively large vector solution space in deblurring networks by optimizing the resultant image I_d. To achieve this, we implemented an end-to-end re-blurring network that processes clear images into blurred images and calculates the mean squared error (MSE) loss of the deblurring results. Directly deducing a blurred image from a clear image is inherently difficult; thus, auxiliary inference using blur information from real blurred images is necessary. Directly inputting clear and blurred images into a heavily blurred network can lead to instability and potential collapse of the training model. To mitigate this, we adopted a simple encoder-decoder architecture inspired by the heavy blur architecture referenced in the literature. This architecture incorporates multiple convolutional, deconvolutional, and ResBlocks (residual blocks). Initially, a blurred image and a clear image are input into the encoder-decoder of the first half of the heavily blurred network. Subsequently, two repeated clear images are input into the second half of the network to adjust the number of channels, allowing the blur features learned in the first half to be effectively transferred to the encoder-decoder of the second half.

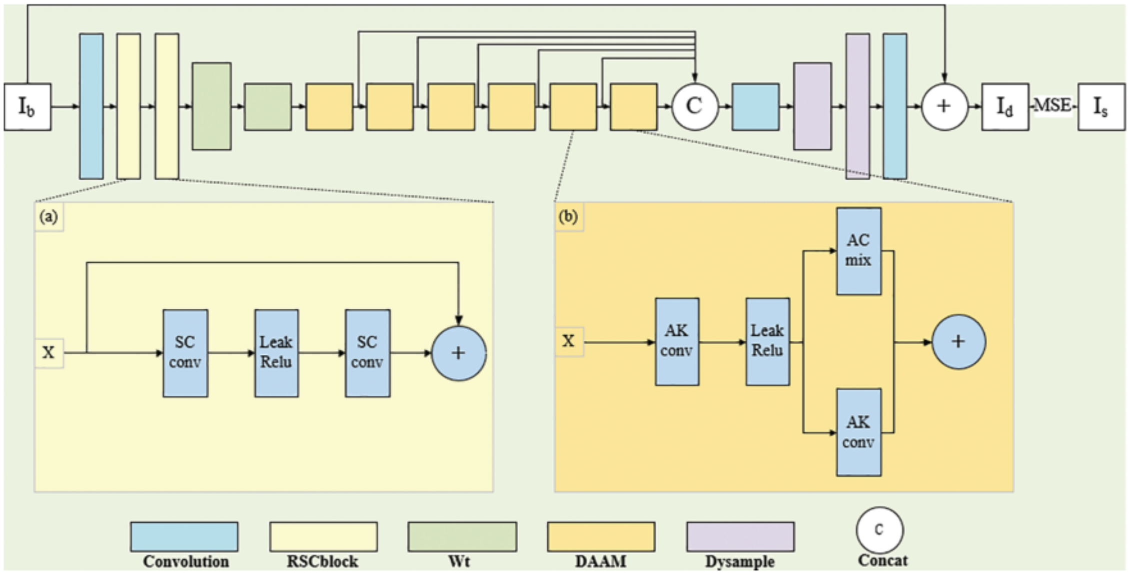

Figure 1: Framework of DeblurringNet

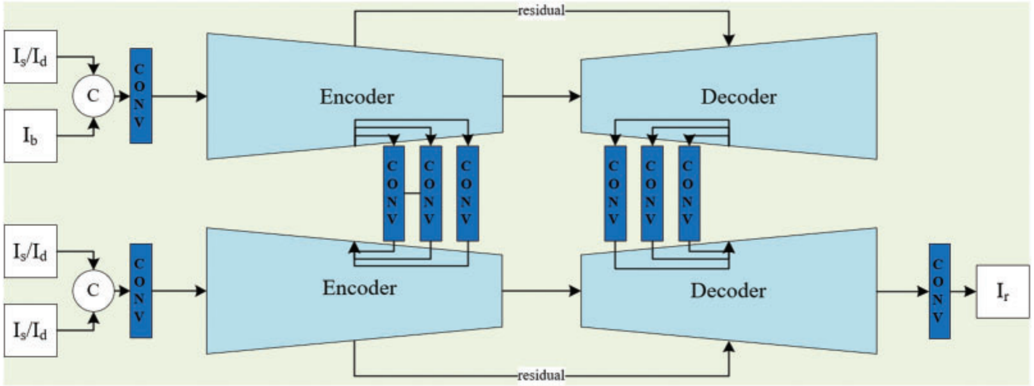

Figure 2: Framework of ReblurringNet

3.2 *Variable Attention Convolution Module*

The DAAM module, illustrated in Fig. 1b, addresses the limitations of standard convolution operations in deep learning-based methods, particularly when applied to pressure-sensitive paint images. Convolutional neural networks (CNNs) are commonly employed to extract feature representations; however, the standard convolution operation, which samples all regions equally, is inadequate for processing these specialized images. This inadequacy arises because the standard convolution operation results in inaccurate light intensity ratio data at pressure holes and marker points in the repaired image. To overcome this issue, we extended the work on deformable convolutions [28]. Typically, a normal convolution kernel has a fixed size, usually K × K, forming a fixed rectangular shape. As the size increases, the number of parameters also increases proportionally, which is problematic because the shape and size of the target can vary across different datasets and locations. Fixed sample shapes and square convolution kernels do not adapt well to these changing targets. Therefore, we introduced akconv, a variable convolution kernel that allows for an arbitrary number of parameters and sample shapes. By incorporating offsets, akconv adjusts the sample shapes at each position, making the convolution process more adaptable. To enhance the flexibility of variable convolution, we established four branches with different expansion rates, generating four offset plots with varying receptive fields.

Attention mechanisms and convolutional operations exhibit a strong correlation. Traditional k × k convolutions can be decomposed into k^2 individual 1 × 1 convolutions, followed by shift and sum operations. In the self-attention module, the projection of queries, keys, and values can similarly be interpreted as multiple 1 × 1 convolutions. Subsequently, attention weights are calculated, and values are aggregated. Thus, the initial phase of both modules involves analogous operations. Notably, this initial phase contributes significantly to the computational complexity, denoted as c^2, in contrast to the second phase. This similarity allows for the integration of the benefits of both convolution and attention mechanisms, while maintaining minimal computational overhead compared to using pure convolution or self-attention alone.

3.3 *Wt Downsampling and Dysample Upsampling*

The wt (wavelet downsampling) method, as depicted in Fig. 3a, was initially applied primarily to semantic segmentation tasks. In conventional Convolutional Neural Networks (CNNs), maximum pooling or convolution with increased step size is typically employed to achieve subsampling and local feature aggregation. However, these methods often result in the loss of spatial information, which is detrimental to pixel-level deblurring. To address this issue, we have replaced the common step convolution with the Haar wavelet downsampling module. The Haar wavelet transform reduces the spatial resolution of feature maps while preserving more semantic information. The structure of this module comprises two main components: a lossless feature coding module and a feature representation learning module. The lossless feature coding module is designed for feature conversion and spatial resolution reduction, effectively reducing the resolution of the feature map while retaining all information. The feature representation learning module, consisting of standard convolution, BN (batch normalization), and ReLU activation, is primarily used to extract discriminative features.

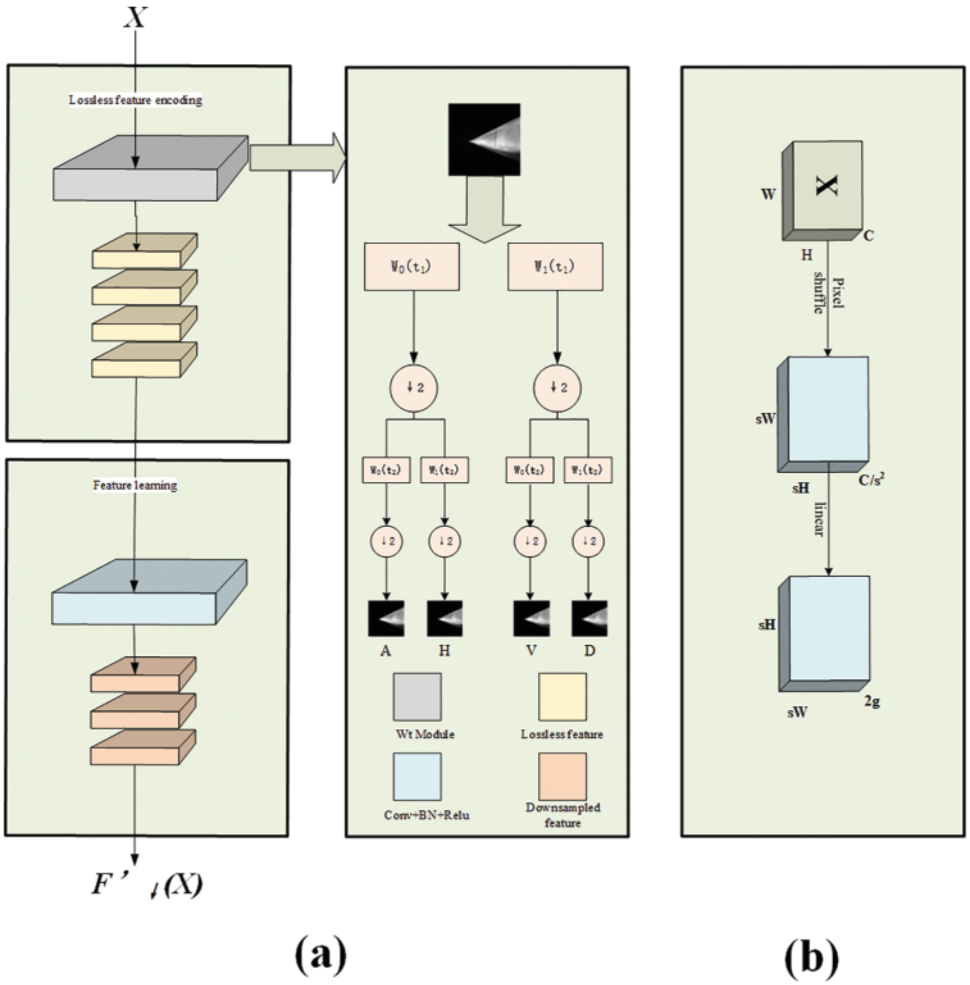

Figure 3: Sampling architecture. (a) Framework of wt, (b) Framework of dysample

During the HWT (Haar wavelet transform) stage, both low-pass and high-pass decomposition filters, denoted as H_0 and H_1, are applied to extract the corresponding low- and high-frequency signals. This operation yields four components: the low-frequency component (A) and three high-frequency components—horizontal (H), vertical (V), and diagonal (D). Each component has half the spatial size of the input, while the number of channels increases by a factor of four. This process effectively encodes spatial information into the channel dimension. In the subsequent feature learning stage, the primary objective is to adjust the channel count of the feature maps to ensure compatibility, while also filtering redundant information. This enables more efficient feature extraction in the downstream layers.

The dysample structure, illustrated in Fig. 3b, restores the resolution of the feature map progressively. Conventional upsampling methods such as nearest neighbor and bilinear interpolation depend on fixed interpolation modes, while techniques like deconvolution and image super-resolution, though popular, are less suitable for the subtle detail recovery needed in fine pressure-sensitive paint imagery. Therefore, we employ dysample upsampling based on point sampling, which does not require high-resolution input features, thereby reducing latency, memory usage, floating-point operations, and parameter count. Moreover, the upsampling position is determined dynamically by learning offsets instead of using fixed interpolation points, providing better adaptability to varied input data. The integration of a controllable scaling factor further enables flexible adjustment of output size by setting different upsampling scales, offering a customizable balance between resolution and efficiency.

3.4 Loss Function

We use the mean square error as the final deblurring loss (Eq. (2)):

$$L_{deblurring} = \| I_s - I_d \|_F^2 \tag{2}$$

Similarly, the loss of a blur network is defined as follows (Eq. (3)):

$$L_{rcblurring} = \|I_r - I_b\|_F^2 \tag{3}$$

After the training convergence of the deblurring and reblurring networks, we replace the I_s in the blur network with the deblurring output I_d, and refine the I_d by deblurring the consistency loss in the blurring. The overall loss function is defined as follows (Eq. (4)):

$$L_{consistency} = L_{deblurring} + \lambda L_{reblurring} \tag{4}$$

where λ is set to 0.1 according to our training experience.

4 Experiments

4.1 Datasets and Metrics

This paper uses the FY dataset provided by the China Aerodynamics Research and Development Center to evaluate the deblurring effect of the DFDNet model on pressure-sensitive paint images. During the acquisition process using the FY standard model to capture pressure-sensitive paint images, both the model and the camera experienced vibrations caused by blowing, which led to non-uniform motion blur. Despite these disturbances, the captured images exhibited high resolution, with rich details and bright textures on the model, as shown in Fig. 4. The dataset consists of 131 training datasets and 39 test datasets.

The GoPro [15] dataset is a classical dataset widely used in image deblurring research. Sharp images captured at 240fps using a GoPro Hero4 Black camera were averaged over time windows of different durations to synthesize blurry images. The clear image in the center of the time window is used as the ground truth image. The GoPro dataset contains 3214 image pairs, divided into a training set and a test set, containing 2103 and 1111 pairs, respectively. This dataset has been extensively utilized to train and evaluate deep deblurring algorithms. The HIDE [16] dataset created a motion-blurred dataset primarily focused on pedestrian and street scenes, incorporating both camera shake and object motion. The dataset contains 6397 and 2025 pairs for training and testing, respectively. Similar to the GoPro dataset, the blurry images in the HIDE dataset are synthesized by averaging 11 process frames, with the middle frame used as the ground truth.

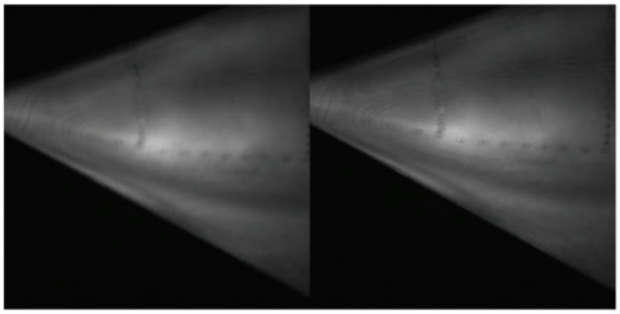

Figure 4: FY picture dataset (left blur, right GT)

In order to evaluate the deblurring effect of DFDNet, it is trained on the FY dataset and GoPro dataset respectively. For the model trained on the FY dataset, we test it on the FY dataset, and for the model trained on the GoPro dataset, we test it on both the GoPro and HIDE datasets.

4.2 Implementation Details

The method is trained by multi-scale supervision on Ubuntu18.04 system on Ubuntu18.04 system with 24G NVIDIA RTX 6000 GPU using PyTorch [29], during training we use Adam [30] optimizer, setup β_1 = 0.5 and β_2 = 0.999, the learning rate is fixed to 0.8×10^{-5} and halved every 1000 epochs. The patch size of all images is cropped to 256 × 256, and the batchsize size is set to 6, and the size of all convolutional filters is 3 × 3. The initial number of channels for all convolutional layers and residual blocks is set to 32 in the deblurring network and 16 in the re-blurring network, and the corresponding number of channels will be doubled or halved in each space.

4.3 Qualitative Comparison

In the FY dataset shown in Fig. 5, focus on the fuselage and nose sections of the flying wing model. DFDNet produced the clearest images, especially of the details in the upper fuselage and the wear in the middle of the fuselage. Marked points at the head site are critical for the accuracy of subsequent pressure settling, and our approach effectively recovers these points and the surrounding details.

In addition, this method is compared with the latest optimal method on the open data set. As shown in. Fig. 6, by analyzing the image of a speeding car in the GOPRO dataset, our method performs better than existing methods in recovering the texture details of the car.

Furthermore, in the HIDE dataset shown in Fig. 7, focus on the fine details of the buildings and advertising columns to the left of the image. DFDNet went beyond competing methods to effectively restore the building's window details and advertising sticker clusters, highlighting its robustness in restoring the structure of complex scenes.

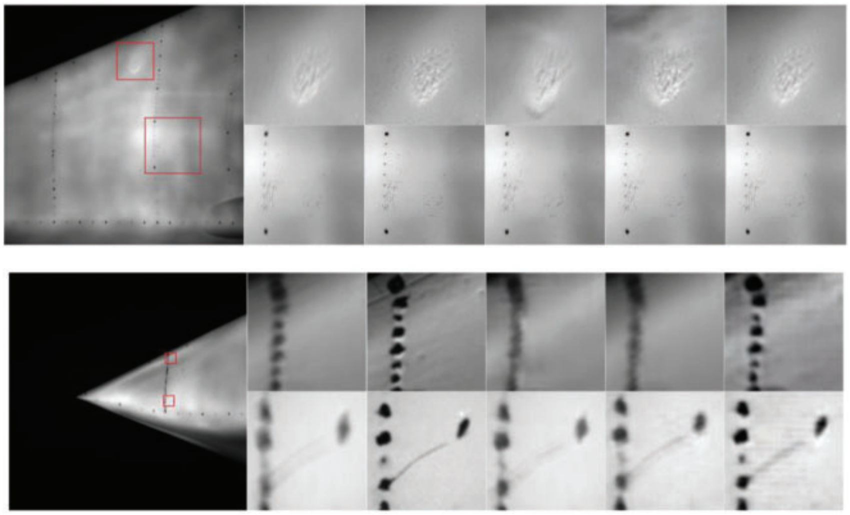

Figure 5: FY dataset test (From left to right, Blur, blur, GT, Stripfomer, NAFNet, FFTformer, and Ours)

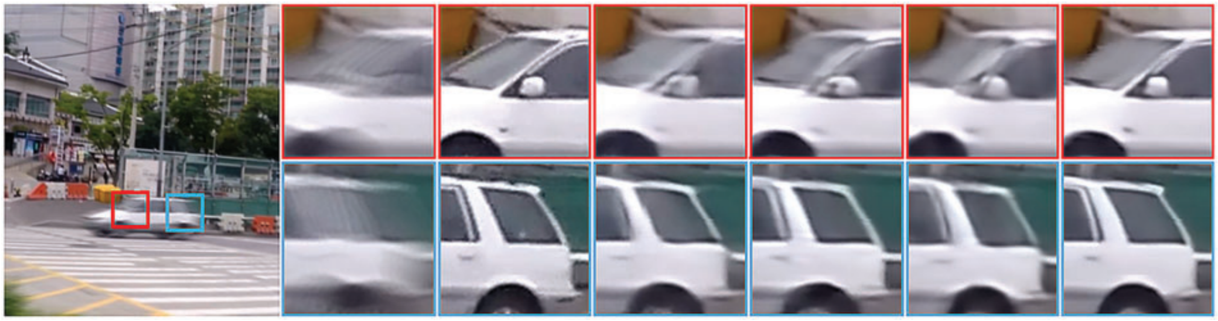

Figure 6: GOPRO dataset test (From left to right, Blur, blur, GT, Stripfomer, NAFNet, FFTformer, and Ours)

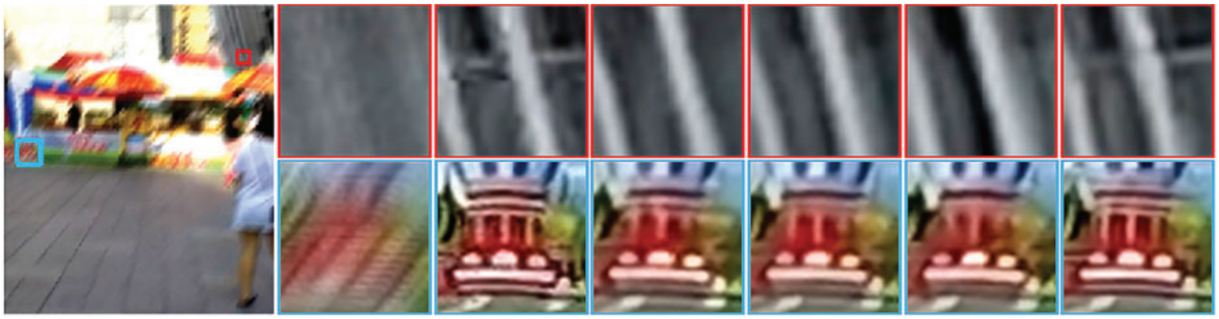

Figure 7: HIDE dataset test (From left to right, Blur, blur, GT, Stripfomer, NAFNet, FFTformer, and Ours)

4.4 Quantitative Comparison

4.4.1 Evaluation of the Effectiveness of the Restoration

In the provided FY dataset, all compared networks were evaluated using pre-trained models on the GOPRO dataset. The performance results are presented in Table 2. Notably, DFDNet exhibited superior performance when applied to pressure-sensitive paint images, which can be attributed to its unique architectural design. The original resolution of these images was 2048 × 2048, which is beyond the supported limit for many existing networks. Therefore, to ensure consistency, all images are uniformly adjusted to a resolution of 1280 × 1280, similar to the resolution in GOPRO, and then fed into the network. The highest-performing results are highlighted in bold for clarity.

Table 2: Quantitative evaluation of FY dataset (PSNR: Peak Signal-to-Noise Ratio, SSIM: Structural Similarity Index Measure)

Method	PSNR↑	SSIM↑
DeblurGANv2 [10]	28.35	0.891
SRN [11]	29.44	0.912
DMPHN [13]	29.88	0.923
MIMO-UNet++3 [31]	30.77	0.931
MPRNet [32]	31.52	0.933
Uformer [33]	31.83	0.931
Restormer [34]	31.55	0.929
Stripformer [35]	31.51	0.932
MSDI-net [36]	31.58	0.929
Restormer+local [37]	32.55	0.938
NAFNet [38]	32.43	0.944
GRL [39]	32.85	0.942
UFPNet [40]	33.42	**0.949**
FFTfomer [41]	33.09	0.948
DFDNet	**33.52**	**0.949**

Additionally, tests were conducted on the GOPRO dataset using 1111 images from its test set, where our method was compared against several state-of-the-art deblurring techniques, as shown in Table 3. The results were obtained directly from the original papers or from official models published by the authors. In Table 3, the PSNR and SSIM results for single-image deblurring methods are reported, along with the parameter counts for each model. The proposed DFDNet was found to achieve the best performance in terms of structural similarity (SSIM) and parameter count (Params), while only a small gap was observed compared to the best-performing FFTformer in peak signal-to-noise ratio (PSNR). The best results are highlighted in bold for emphasis.

For the HIDE dataset, 2025 images from its test set were used to evaluate the performance, and the results are presented in Table 4. The same deblurring strategy was employed as with the GOPRO dataset. Commendable results were achieved by our network on this dataset as well, demonstrating its strong generalization ability. Notably, these results were obtained without any specific adaptation using the HIDE training set, relying solely on pre-training with the GOPRO dataset. The highest-performing results are highlighted in bold font.

Table 3: Quantitative evaluation of GOPRO datasets (Params: Represents the total number of all trainable parameters in the model)

Method	PSNR↑	SSIM↑	Params (M)↓
DeblurGANv2	29.55	0.934	60.9
SRN	30.26	0.934	6.8
DMPHN	31.36	0.947	21.8
MIMO-UNet++	32.45	0.956	15.8
MPRNet	32.66	0.958	20
Uformer	33.06	0.967	50.9
Restormer	32.92	0.961	26.1
Stripformer	33.08	0.962	26.1
MSDI-net	33.28	0.964	135.4
Restormer+local	33.57	0.965	26.1
NAFNet	33.71	0.966	67.9
GRL	33.93	0.968	20.2
UFPNet	34.06	0.968	80.3
FFTfomer	**34.21**	**0.969**	16.6
DFDNet	34.01	0.968	**5.29**

Table 4: Quantitative evaluation of HIDE datasets

Method	PSNR↑	SSIM↑
DeblurGANv2	26.61	0.875
SRN	28.36	0.904
DMPHN	29.1	0.918
MIMO-UNet++	29.99	0.93
MPRNet	30.96	0.939
Uformer	30.9	0.953
Restormer	31.22	0.942
Stripformer	31.03	0.933
MSDI-net	31.02	0.94
Restormer+local	31.49	0.944
NAFNet	31.31	0.942
GRL	31.65	**0.947**
UFPNet	**31.74**	**0.947**
FFTfomer	31.62	0.945
DFDNet	31.64	0.944

4.4.2 FY Image Deblurring Accuracy Evaluation

To verify the accuracy of restoring the true pressure of the pressure-sensitive paint after deblurring in practical applications, previously provided blurred images, deblurred images, and clear images were utilized for evaluation. These images were collected under conditions of 95 KPa pressure, 30°C temperature, and 60 m/s wind speed. The pressure distribution image was derived through standardized preprocessing, including marking point positioning, registration, and pressure calculation after hole repair. The image was divided into the first, second, and third sections from left to right based on pre-calibrated hole positions. In Fig. 8, the *in-situ* Marking fitting result b is illustrated by importing the true pressure values into the hole positions of the first, second, and third sections in the clear image. When the upper and lower pressure limits were set at 89,000 and 95,000 Pa, respectively, it became evident that the pressure distribution at the pressure measurement holes was positively correlated with the line obtained by fitting the points corresponding to the light intensity ratio in our image. This also corresponds to the pressure sensitivity curve obtained through pre-static calibration of the pressure-sensitive paint.

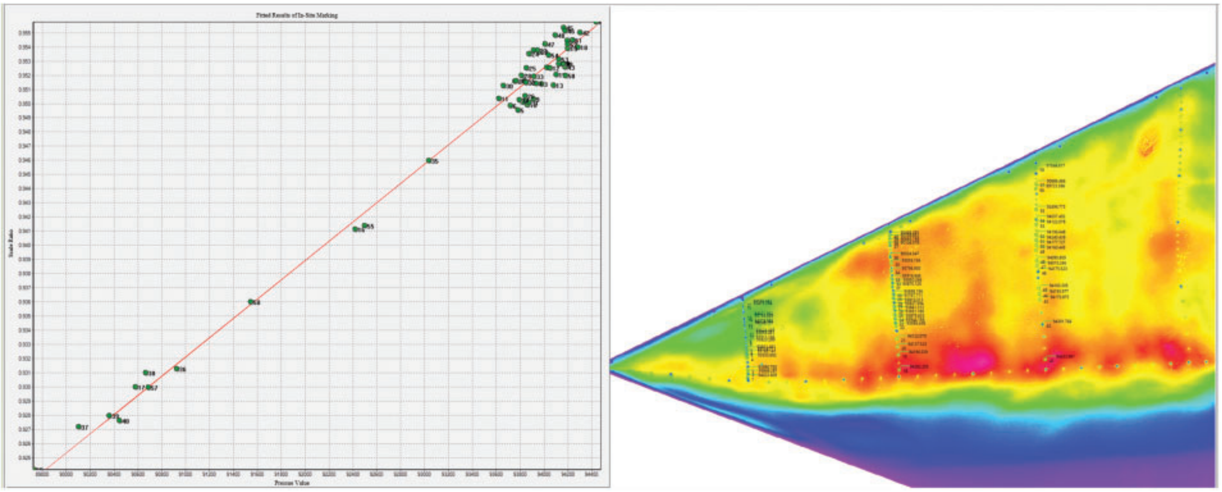

Figure 8: Real pressure values and fitted results of *in-situ* Marking on the first, second and third sections of the clear image

Fig. 9 demonstrates that the pressure distribution of image (b), after being deblurred by DFDNet, closely resembles that of the clear image (a). In contrast, the original image (c), without deblurring, exhibits significant pressure fluctuations. It is observed that, after DFDNet processing, the gray value of the surrounding pixels can approximate the true pressure value, regardless of whether the region lies along the edge, in the high-pressure area, or in the low-pressure area on the lower side.

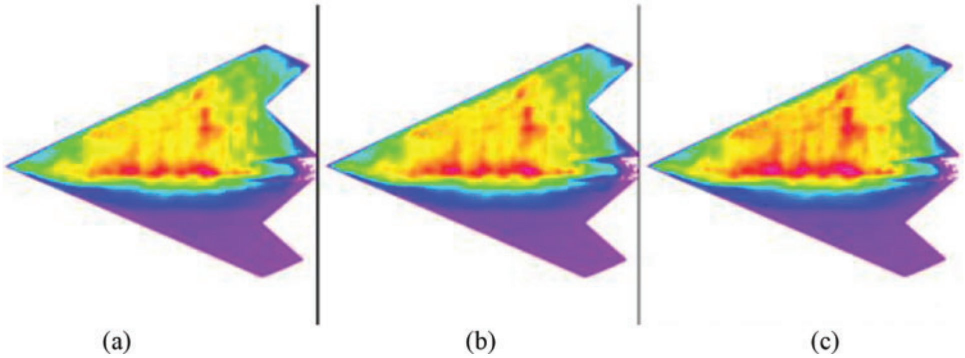

Figure 9: FY pressure distribution (from left to right (a) clear, (b) DFDNet, (c) FFTformer)

To accurately evaluate the pressure distribution image after deblurring, we used the pressure data from the real clear image as the benchmark. We then compared the pressure data after deblurring with the image without deblurring to derive the average relative error formula (Eq. (5)):

$$MRE = \frac{1}{N}\sum\nolimits_{i=1}^{N}\left|\frac{p_{1,i} - p_{2,i}}{p_{1,i}}\right| \tag{5}$$

In this context, N represents the total number of pixels in the image, while $p_{1,i}$ and $p_{2,i}$ denote the pressure values of the i-th pixel in the first and second images, respectively. The formula calculates the relative error of the pressure value for each pixel between the two images. Specifically, it computes the difference between the pressure values of the corresponding pixels, divides this difference by the absolute value of the pressure value in the first image, and then takes the absolute value of the resulting relative error. This process is repeated for all pixels, and the sum of these absolute relative errors is then divided by the total number of pixels to obtain the average relative error. A smaller average relative error indicates a smaller difference between the pixel pressure values in the two images, thereby signifying a higher similarity.

Upon calculation, as shown in Table 5, it was found that the average relative error after DFDNet restoration is closer to the pressure value with minimal fluctuation. This result demonstrates that the proposed network model can effectively mitigate the imbalance in image light intensity ratios caused by blur. Consequently, it aids in restoring image accuracy, laying the foundation for subsequent experiments.

Table 5: Ablation experiments performed on the FY dataset

Method	DFDNet	FFTformer
MRE↓	**0.0907**	0.0957

4.5 Ablation Experiment

To objectively evaluate the contribution of each module within DFDNet, ablation experiments were conducted by varying the quantities and combinations of the modules. For verification, 100 images were randomly selected from the GOPRO dataset. All versions of the network were trained using the previously established loss functions and settings. The ablation results for the DAAM modules are presented in Table 6. The results indicate that deblurring performance improves significantly as DAAM modules are incrementally stacked, achieving optimal performance with six DAAM modules. Beyond this point, further increasing the number of DAAM modules leads to a decline in performance; simply adding more modules does not enhance performance but instead increases the parameter count non-linearly.

To further compare the effectiveness of each module, additional experiments were conducted using six DAAM modules, which were successively replaced with baseline components, such as ordinary convolution downsampling, deconvolution upsampling, and two ResBlocks. Specifically, ordinary convolution downsampling was replaced by “wt,” ResBlock was replaced by “RSCblock,” and deconvolution was replaced by “dysample.” The results of these experiments are shown in Table 7. The findings reveal that upgrading the model from the baseline configuration underscores the critical role of the RSCblock in enhancing the deblurring effect. This suggests that attention mechanisms in smaller receptive fields are more focused on static objects with less motion, whereas attention mechanisms in larger receptive fields are more sensitive to dynamic objects with greater motion.

Table 6: Ablation experiments performed on the GOPRO dataset

Method	PSNR↑	SSIM↑	Params (M)↓
DAAM*2	29.84	0.915	4.12
DAAM*4	32.45	0.952	4.75
DAAM*6	**34.19**	**0.969**	**5.29**
DAAM*8	33.98	0.966	7.58
DAAM*10	33.55	0.962	12.37

Table 7: Quantitative evaluation of FY dataset

Method	PSNR↑	SSIM↑
Baseline	30.95	0.935
Wt	31.52	0.935
RSCblock	31.23	0.935
RSCblock*2	32.67	0.941
Dysample	31.38	0.937
Wt+RSCblock	32.86	0.943
Wt+RSCblock*2	33.04	0.944
Wt+RSCblock*2+dysample	**33.52**	**0.949**

5 Conclusion

In conclusion, DFDNet, a variable attention convolutional network, was presented in this study to address the issue of motion blur in high-resolution pressure-sensitive paint images obtained from wind tunnel experiments. DFDNet leverages DAAM and RSCBlock to effectively mitigate non-uniform motion blur. It demonstrates superior performance on the FY dataset and achieves competitive results on established benchmarks, including GOPRO and HIDE. Despite its commendable generalization capabilities, DFDNet exhibits certain limitations. The complexity inherent in the model, due to the incorporation of multiple specialized modules, results in heightened computational demands, which may limit its applicability in real-time or resource-constrained environments. Furthermore, while the model is optimized for pressure-sensitive paint images, its effectiveness on other specialized image types, such as low-light or medical images, remains untested and may require further optimization. Future work will focus on enhancing the efficiency of the model by exploring lightweight alternatives and expanding its application to other complex imaging domains. Additionally, the incorporation of domain adaptation techniques could further enhance the generalizability of DFDNet, enabling its broader utilization across diverse imaging fields.

Acknowledgement: The authors express gratitude to all institutions and teachers who have provided assistance with this research.

Funding Statement: This project is supported by the National Natural Science Foundation of China (No. 12202476).

Author Contributions: Network architecture design: Ruizhe Yu and Zhisheng Gao; Data collection: Tingrui Yue; Result analysis: Analysis by Ruizhe Yu and Lei Liang, draft written by Ruizhe Yu. All authors reviewed the results and approved the final version of the manuscript.

Availability of Data and Materials: If you need FY dataset, please contact gzs_xihua@xhu.edu.cn.

Ethics Approval: Not applicable.

Conflicts of Interest: The authors declare no conflicts of interest to report regarding the present study.

References

1. Gregory JW, Disotell KJ, Peng D, Juliano TJ, Crafton J, Komerath NM. Inverse methods for deblurring pressure-sensitive paint images of rotating surfaces. AIAA J. 2014 Sep;52:2045–61. doi:10.2514/1.J052793.
2. Juliano TJ, Disotell KJ, Gregory JW, Crafton J, Fonov S. Motion-deblurred, fast-response pressure-sensitive paint on a rotor in forward flight. Meas Sci Technol. 2012 Apr;23:045303. doi:10.1088/0957-0233/23/4/045303.
3. Gong D, Yang J, Liu L, Zhang Y, Reid I, Shen C, et al. From motion blur to motion flow: a deep learning solution for removing heterogeneous motion blur. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR); 2017 Jul; Honolulu, HI, USA. p. 2319–28.
4. Szeliski R. Computer vision: algorithms and applications. In: Choice reviews online. Berlin, Germany: Springer; 2011 May. p. 48-5140.
5. Xu L, Tao X, Jia J. Inverse kernels for fast spatial deconvolution. In: Computer vision–ECCV 2014, lecture notes in computer science. Cham, Switzerland: Springer; 2014. p. 33–48.
6. Bahat Y, Efrat N, Irani M. Non-uniform blind deblurring by reblurring. In: 2017 IEEE International Conference on Computer Vision (ICCV); 2017 Oct; Venice, Italy. p. 3286–94.
7. Fergus R, Singh B, Hertzmann A, Roweis ST, Freeman WT. Removing camera shake from a single photograph. ACM Transact Grap. 2006 Jul;25:787–94. doi:10.1145/1141911.1141956.
8. Xu L, Jia J. Two-phase kernel estimation for robust motion deblurring. In: Computer vision–ECCV 2010, lecture notes in computer science. Berlin, Germany: Springer; 2010. p. 157–70.
9. Kupyn O, Budzan V, Mykhailych M, Mishkin D, Matas J. DeblurGAN: blind motion deblurring using conditional adversarial networks. In: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition; 2018 Jun; Salt Lake City, UT, USA. p. 8183–92.
10. Kupyn O, Martyniuk T, Wu J, Wang Z. DeblurGAN-v2: deblurring (Orders-of-Magnitude) faster and better. In: 2019 IEEE/CVF International Conference on Computer Vision (ICCV); 2019 Oct; Seoul, Republic of Korea. p. 8878–87.
11. Tao X, Gao H, Shen X, Wang J, Jia J. Scale-recurrent network for deep image deblurring. In: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition; 2018 Jun; Salt Lake City, UT, USA. p. 8174–82.
12. Yuan Y, Su W, Ma D. Efficient dynamic scene deblurring using spatially variant deconvolution network with optical flow guided training. In: 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR); 2020 Jun; Seattle, WA, USA. p. 3555–64.
13. Zhang H, Dai Y, Li H, Koniusz P. Deep stacked hierarchical multi-patch network for image deblurring. In: 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR); 2019 Jun; Long Beach, CA, USA. p. 5978–86.
14. Zhang J, Pan J, Ren J, Song Y, Bao L, Lau RWH, et al. Dynamic scene deblurring using spatially variant recurrent neural networks. In: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition; 2018 Jun; Salt Lake City, UT, USA. p. 2521–9.
15. Nah S, Kim TH, Lee KM. Deep multi-scale convolutional neural network for dynamic scene deblurring. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR); 2017 Jul; Honolulu, HI, USA; p. 3883–91.
16. Shen Z, Wang W, Lu X, Shen J, Ling HB, Xu TF, et al. Human-aware motion deblurring. In: 2019 IEEE/CVF International Conference on Computer Vision (ICCV); 2019 Oct; Seoul, Republic of Korea. p. 5572–81.
17. Gao H, Tao X, Shen X, Jia J. Dynamic scene deblurring with parameter selective sharing and nested skip connections. In: 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR); 2019 Jun; Long Beach, CA, USA. p. 3848–56.

18. Cai J, Zuo W, Zhang L. Dark and bright channel prior embedded network for dynamic scene deblurring. IEEE Transact Image Process. 2020 Jan;29:6885–97. doi:10.1109/TIP.2020.2995048.
19. Suin M, Purohit K, Rajagopalan AN. Spatially-attentive patch-hierarchical network for adaptive motion deblurring. In: 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR); 2020 Jun; Seattle, WA, USA. p. 3606–15.
20. Zamir SW, Arora A, Khan S, Hayat M, Khan FS, Yang MH, et al. Multi-stage progressive image restoration. In: 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR); 2021 Jun; Nashville, TN, USA. p. 14821–14831.
21. Mao X, Liu Y, Shen W, Li Q, Wang Y. Deep residual fourier transformation for single image deblurring. arXiv:2111.11745. 2021.
22. Mao X, Liu Y, Liu F, Li Q, Shen W, Wang Y. Intriguing findings of frequency selection for image deblurring. Proc AAAI Conf Artif Intell. 2023;37(2):1905–13.
23. Zhou C, Teng M, Han J, Liang J, Xu C, Cao G, et al. Deblurring low-light images with events. Int J Comput Vis. 2023 May;131:1284–98. doi:10.1007/s11263-023-01754-5.
24. Lee HS, Cho SI. Locally adaptive channel attention-based spatial-spectral neural network for image deblurring. IEEE Transact Circ Syst Video Technol. 2023 Oct;33:5375–90. doi:10.1109/TCSVT.2023.3250509.
25. Whyte O, Sivic J, Zisserman A, Ponce J. Non-uniform deblurring for shaken images. In: 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition; 2010 Jun; San Francisco, CA, USA. p. 168–86.
26. Harmeling S, Hirsch M, Schölkopf B. Space-variant single-image blind deconvolution for removing camera shake. In: Advances in neural information processing systems. Red Hook, NY, USA: Curran Associates, Inc.; 2010. vol. 23, p. 829–37.
27. Li J, Wang W, Nan Y, Ji H. Self-supervised blind motion deblurring with deep expectation maximization. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. Los Alamitos, CA, USA: IEEE Computer Society; 2023. p. 1234–43.
28. Zhu X, Hu H, Lin S, Dai J. Deformable ConvNets v2: more deformable, better results. In: 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR); 2019 Jun; Long Beach, CA, USA. p. 9308–16.
29. Paszke A, Gross S, Massa F, Lerer A, Bradbury J, Chanan G, et al. PyTorch: an imperative style, high-performance deep learning library. In: Neural Information Processing Systems, Neural Information Processing Systems. Red Hook, NY, USA: Curran Associates, Inc.; 2019 Jan.
30. Kingma DP, Ba J. Adam: a method for stochastic optimization. 2014 Dec. doi:10.48550/arXiv.1412.6980.
31. Cho S-J, Ji S-W, Hong J-P, Jung S-W, Ko S-J. Rethinking Coarse-to-fine approach in single image deblurring. In: 2021 IEEE/CVF International Conference on Computer Vision (ICCV); 2021 Oct; Montreal, QC, Canada.
32. Mehri A, Ardakani PB, Sappa AD. MPRNet: multi-path residual network for lightweight image super resolution. In: 2021 IEEE Winter Conference on Applications of Computer Vision (WACV); 2021 Jan; Waikoloa, HI, USA.
33. Wang Z, Cun X, Bao J, Zhou W, Liu J, Li H. Uformer: a general U-shaped transformer for image restoration. In: 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR); 2022 Jun; New Orleans, LA, USA.
34. Zamir SW, Arora A, Khan S, Hayat M, Khan FS, Yang M-H. Restormer: efficient transformer for high-resolution image restoration. In: 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR); 2022 Jun; New Orleans, LA, USA.
35. Tsai F-J, Peng Y-T, Lin Y-Y, Tsai C-C, Lin C-W. StripFormer: strip transformer for fast image deblurring. In: European Conference on Computer Vision. Cham, Switzerland: Springer; 2022 Apr. p. 146–62.
36. Li D, Zhang Y, Cheung KC, Wang X, Qin H, Li H. Learning degradation representations for image deblurring. In: European Conference on Computer Vision. Cham, Switzerland: Springer; 2022 Oct. p. 736–53.
37. Chu X, Chen L, Chen C, Lu X. Improving image restoration by revisiting global information aggregation. In: Computer Vision–ECCV 2022, lecture notes in computer science. Cham, Switzerland: Springer; 2022. p. 53–71.
38. Chen L, Chu X, Zhang X, Sun J. Simple baselines for image restoration. In: European Conference on Computer Vision. Cham, Switzerland: Springer; 2022 Apr. p. 17–33.

39. Li Y, Fan Y, Xiang X, Demandolx D, Ranjan R, Timofte R, et al. Efficient and explicit modelling of image hierarchies for image restoration. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. Los Alamitos, CA, USA: IEEE Computer Society; 2023 Mar. p. 18278–89.
40. Fang Z, Wu F, Dong W, Li X, Wu J, Shi G. 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. Los Alamitos, CA, USA: IEEE Computer Society; 2023. p. 18105–14.
41. Kong L, Dong J, Li M, Ge J, Pan J. Efficient frequency domain-based transformers for high-quality image deblurring. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. Los Alamitos, CA, USA: IEEE Computer Society; 2022 Nov. p. 5886–95.