Article

A Lightweight Fusion Distillation Network for Image Deblurring and Deraining [†]

Yanni Zhang ^{1,2,‡} , Yiming Liu ^{3,‡}, Qiang Li ¹, Jianzhong Wang ¹, Miao Qi ^{1,2}, Hui Sun ², Hui Xu ^{2,*} and Jun Kong ^{2,*}

¹ College of Information Science and Technology, Northeast Normal University, Changchun 130000, China; zhangyn500@nenu.edu.cn (Y.Z.); liq782@nenu.edu.cn (Q.L.); wangjz019@nenu.edu.cn (J.W.); qim801@nenu.edu.cn (M.Q.)

² Institute for Intelligent Elderly Care, Changchun Humanities and Sciences College, Changchun 130000, China; sunh333@nenu.edu.cn

³ College of Information Science and Engineering, Hunan Normal University, Changsha 410000, China; yimingliu321@163.com

* Correspondence: xuh504@nenu.edu.cn (H.X.); kongjun@nenu.edu.cn (J.K.)

[†] An earlier version of this paper was presented at the International Conference on the 28th IEEE International Conference on Image Processing (IEEE ICIP 2021). Zhang, Y.; Liu, Y.; Li, Q.; Qi, M.; Xu, D.; Kong, J.; Wang, J. Image deblurring based on lightweight multi-information fusion network. CoRR 2021, abs/2101.05403, [2101.05403].

[‡] These authors contributed equally to this work.

Abstract: Recently, deep learning-based image deblurring and deraining have been well developed. However, most of these methods fail to distill the useful features. What is more, exploiting the detailed image features in a deep learning framework always requires a mass of parameters, which inevitably makes the network suffer from a high computational burden. We propose a lightweight fusion distillation network (LFDN) for image deblurring and deraining to solve the above problems. The proposed LFDN is designed as an encoder–decoder architecture. In the encoding stage, the image feature is reduced to various small-scale spaces for multi-scale information extraction and fusion without much information loss. Then, a feature distillation normalization block is designed at the beginning of the decoding stage, which enables the network to distill and screen valuable channel information of feature maps continuously. Besides, an information fusion strategy between distillation modules and feature channels is also carried out by the attention mechanism. By fusing different information in the proposed approach, our network can achieve state-of-the-art image deblurring and deraining results with a smaller number of parameters and outperform the existing methods in model complexity.

Keywords: image deblurring; image deraining; lightweight distillation block; fusion mechanism

Citation: Zhang, Y.; Liu, Y.; Li, Q.; Wang, J.; Qi, M.; Sun, H.; Xu, H.; Kong, J. A Lightweight Fusion Distillation Network for Image Deblurring and Deraining. *Sensors* **2021**, *21*, 5312. <https://doi.org/10.3390/s21165312>

Academic Editor: Zahir M. Hussain

Received: 17 June 2021

Accepted: 4 August 2021

Published: 6 August 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Image deblurring and deraining are both typical and essential tasks in image restoration research. Blurry images are caused by the movement of objects or camera shaking, and rainy images often distort the color of the background image due to the blocking and refraction of sunlight by rain. Both of these affect the vision sensors and people's visual perception of detailed image information. Therefore, image deblurring and deraining have become indispensable steps for many computer vision tasks, such as object detection, image classification, and surveillance. However, estimating the blur kernel and rain streaks to restore sharp and clean images from the blurry and rainy ones is difficult since it is a highly ill-posed problem. Most traditional methods [1–6] regularize the solution space by introducing some prior information to model the blur kernel and rain streaks. However, because the prior information may not conform to the real scene, the quality of the restored images is not optimal. Recently, some information fusion based methods have also been proposed for image restoration problem. For instance, Zhu [7] utilized a set of artificial

images to construct pixelwise weight maps, so that both global and local image features can be analyzed and fused for image dehazing. In [8], a multiple image fusion scheme based on adaptive structural decomposition was also proposed for haze image enhancement. While these information fusion methods can avoid prior information estimation and restore the image without inverting the image hazing model, they still belong to the shallow image restoration model. Therefore, the methods in [7,8] rely on carefully designed filter, criterion or measure to characterize the image features.

With the rapid development of deep learning, many convolutional neural network (CNN) based methods [9–22] have been proposed for effective image deblurring [9–18,23] and deraining [19,22,24–28]. Compared with the traditional shallow models [1–6], these deep CNN methods do not need to estimate blur kernel or rain streaks, but directly predict clear images from the degradation ones. Furthermore, they also avoid careful engineering to design image feature extractor. While these methods are effective, they still have two shortcomings. First, most of the existing deep CNN methods cannot distillate the useful features from all extracted features, so some useless feature information will interfere with the network and prevent the model from recovering the details of the image. While some recent studies attempt to solve this problem by employing the attention mechanism [19,23,25,27,29], they still have drawbacks. MPRNet [25] added a supervised attention module (SAM) between every two stages to enable progressive learning. However, only the local image information was focused in SAM. Some deraining [19,27] and deblurring [23,29] methods adopted the channel attention mechanism [30] to filter useful feature. However, the channel attention mechanism [30] just assigned different weights to different feature maps rather than reorganize useful information. The second problem of the existing deblurring and deraining methods is the complexity of the network model. In order to take full advantage of the surrounding pixels to recover image details, multi-scale methods [9–13,25] are commonly adopted in the image deblurring and deraining task. While this multi-scale strategy can obtain different receptive fields at different scales, it sacrifices the complexity of the network. Nah et al. [9] introduced three kinds of blurry images with different sizes into the model, which required the network to carry out feature extraction and image reconstruction three times. Therefore, the number of network parameters is too large. SRN [10] employed the LSTM mechanism to share feature extraction results across scales, but it still failed to solve the problem fundamentally. Zhang et al. [13], DeblurGANv2 [16], PReNet [22], MPRNet [25] also tackled the deraining and deblurring problem in multiple stages, which inevitably leads to the increase of network parameters.

This paper proposes a lightweight fusion distillation network (LFDN) to address the problems mentioned above. Compared with the existing methods, our LFDN could successfully recover the image details from the blurry and rainy images with a lightweight network structure, as shown in Figure 1. The contribution can be summarized as follows:

- We propose a multi-scale hierarchical information fusion scheme (MSHF) to encode the image with rain and blur. MSHF extracts and fuses the image feature in multiple small-scale spaces, which can eliminate redundant parameters while maintaining the rich image information.
- We propose a very lightweight module named feature distillation normalization block (FDNB) which can constantly filter out useless feature channel information. To the best of our knowledge, it is the first time that the distillation network is adopted in image deblurring and deraining tasks.
- Two attention mechanism based modules are also presented in the decoding process of our approach to exploit the interdependency between the layers and feature channels, which is termed as Multi-feature fusion module based on attention mechanism (MFFD). Through MFFD, a better information fusion can be achieved to compensate for the potential image detail lost in FDNB.

Figure 1. The PSNR and average inference time of state-of-the-art deep learning based deblurring and deraining methods on GoPro and Rain14000 Datasets. Our proposed LFDN achieves the best performance with the least time. (Blue and red marks represent the methods for deblurring and deraining, respectively.)

2. Related Work

Image Deblurring. In recent years, CNNs based image deblurring has been developed rapidly. Nah et al. [9] proposed a multi-scale CNN method for image deblurring called DeepDeblur, which is based on a coarse-to-fine strategy to restore the sharp image progressively. While it achieved satisfactory results, DeepDeblur contains 40 convolutional layers in each scale without parameter sharing. Zhang et al. [13] proposed a spatial variant neural network that consists of three CNNs and a recurrent neural network (RNN) for dynamic scene deblurring. While this algorithm is effective, it uses many convolutional layers for feature extraction and weights estimation by RNNs, which increase the number of parameters. Tao et al. [10] employed an encoder–decoder structure to propose a scale-recurrent network (SRN). Compared with DeepDeblur, SRN applied the long-short term memory (LSTM) to share weights across scales. Similarly, the parameter sharing scheme is also adopted by Gao et al. [11] to improve the efficiency of their image deblurring network. Zhang et al. [12] investigated a new scheme that exploits the deblurring cues at different scales via a hierarchical multi-patch model and proposed a simple yet effective multi-level CNNs model called Deep Multi-Patch Hierarchical Network (DMPHN). Nevertheless, under the coarse-to-fine scheme, most networks use a large number of training parameters due to large filter sizes. Thus, the multi-scale, scale-recurrent, and multi-patch methods result in an expensive runtime and struggle to improve deblurring quality. In order to solve this problem and further make use of multi-scale mechanisms, Kupyn et al. [16] introduced a feature pyramid network (FPN) [31] to replace the multi-scale strategy, which can handle both semantic information and context detail of the blurry image. However, the problem of semantic information dilution occurs in its decoding process.

Image Deraining. Deng et al. [18] constructed a specialized detail repair network that utilizes some well-designed structure detail context aggregation blocks (SDCAB) for image deraining. However, SDCAB only focused on local feature fusion, which may lead its result to be suboptimal. Jiang et al. [19] employed a recurrent manner to capture the global image texture, thus allowing to explore the complementary and redundant information at the spatial dimension to characterize target rain streaks. However, this method required both rainy image and rain strike information as the input of network. Thus, it cannot be applied to most benchmark datasets which only contain rainy images without the rain strike information. Recently, an uncertainty guided multi-scale residual learning network (UMRL) [28] was proposed to learn the rain pattern information at different scales. At the same time, a cycle spinning frame was also used to remove artifacts. While UMRL achieved

efficient results, it needs three kinds of images at different scales as input, which inevitably increases the model complexity. Ren et al. [22] provided a better and simpler baseline deraining network by simultaneously considering network architecture, input, output, and loss functions. MPRNet [25] proposed a multi-stage architecture that progressively learns restoration functions for the degraded inputs. Specifically, MPRNet adopted the encoder–decoder architecture to learn context-specific features and then combined them with high-resolution branching to retain local information. However, MPRNet ignored the global information. RESCAN [27] defined heavy rain as the accumulation of multiple rainwater layers. Because rainwater stripe layers overlap each other, it is not easy to remove rainwater completely in a single stage. Therefore, they proposed a deep network combined with a recursive neural network that preserves useful information in the first stage and then facilitates the removal of rainwater in the second stage. However, the multi-stage strategy makes RESCAN suffer from the problem of model complexity.

Attention Mechanism. Inspired by its successful applications in natural language processing, the attention mechanism has been widely used in image processing tasks [19,25,29,32–35]. Zhang et al. [32] leveraged the attention mechanism to allow the network to focus on the relationship among spatial image areas. Kuldeep et al. [29] utilized the weighted sum of all location features to selectively aggregate the location feature. Kim et al. [33] learned the correlation between feature channels through residual blocks and spatial channel attention. MSPFN [19] combined pyramid structure [31] and channel attention mechanism [30] to synergistically represent multi-scale rain-pattern information. More recently, Niu et al. [35] proposed a holistic attention network (HAN) to characterize the relationship between network layers. Nevertheless, HAN regarded multiple feature channels of each layer as a whole group and only focused on estimating the correlation between feature channel groups of different layers. Thus, the correlation between the inter-layer feature channels was ignored. Moreover, the 3D convolution in HAN increased the number of parameters and computational burden dramatically.

Distillation Network. The information distillation network is one of the state-of-the-art methods to reduce the number of parameters and achieve a lightweight network architecture. Zheng et al. [36] proposed an information distillation network (IDN) which reduces the computational complexity and memory consumption by channel splitting strategy to downscale the feature maps. Based on IDN, a fast and lightweight information multi-distillation network (IMDN) [37] was presented. IMDN extracted features at a granular level by applying the channel splitting strategy multiple times and proposed contrast-aware channel attention (CCA) to connect the extracted features. However, Liu et al. pointed out that IMDB is still inflexible and inefficient [38]. As a result, they introduced the residual feature distillation block (RFDB), which utilizes the feature distillation connection instead of the channel splitting strategy. Thus, RFDB can improve its performance without introducing additional parameters. While the distillation network has already been employed in some computer vision problems, there are few studies adopt it for image deblurring and deraining. Furthermore, the hierarchical information of different distillation layers is also neglected in the existing methods. Finally, as shown in StyleGAN [39], normalization is very important for low-level vision tasks, but none of the existing distillation methods contains a normalization layer.

3. Proposed Method

3.1. Overview

The overall architecture of the proposed LFDN is shown in Figure 2, where the green block represents multi-scale hierarchical fusion module (MSHF), red block represents pixel-shuffle up-sampling, and $\oplus$ represents the operation of elementwise addition. Our LFDN is based on an encoder–decoder structure and consists of two main parts. The feature of blurry or rainy image is firstly extracted by a convolution layer with kernel size as 3×3 and step size as 1. Then the extracted feature map is input into the multi-scale hierarchical fusion module (MSHF) in Part I for down-sampling, which acts as the encoder of our model.

Through MSHF, the input will be encoded into several small-scale features, and the loss of information will be compensated by the information fusion between different layers. Part II is the multi-feature fusion distillation module (MFFD), which includes several feature distillation normalization blocks (FDNBs) and two attention fusion mechanisms. This part makes the network have the ability to screen useful features by a finer-grained feature extraction strategy while reducing the number of parameters. After part II, it comes to the process of decoder, which combines Residual Convolution block [40] and pixel-shuffle up-sampling block [41] to enlarge the size of feature map to obtain restored image.

Figure 2. The architecture of lightweight fusion distillation network (LFDN). Part I is the multi-scale hierarchical fusion module (MSHF), Part II is the multi-feature fusion distillation module (MFFD) contains three types of blocks: FDNB is the block of feature distillation normalization, ALFM is the attention layer fusion module, ACFM is the attention channel fusion module. By Part I and Part II, the clear image with more detailed information can be obtained with less network parameters.

3.2. Multi-Scale Hierarchical Information Fusion Scheme (MSHF)

As shown in Figure 3, MSHF is a down-sampling module that serves as the encoder in our approach. First, we denote the input feature map as f_0 , and its size is $W \times H \times C$. Then, the intermediate features f_i ($i = 1, 2, 3, 4$) of f_0 are progressively extracted using downblock modules [34]. The downblock module consists of two convolution layers, one has a kernel size of 3×3 , a step size of 2 to make the feature map down-sampled, and the other has a kernel size of 3×3 , a step size of 1 to make the feature map resampled. Each downblock module [34] will reduce the size of the feature by half and the resampling in it will further refine the feature. After each downblock module for down-sampling, we simply use resblock [40] to realize residual learning. In addition, we fuse different scale features in the small-scale space. The small-scale features after residual learning are up-sampled by pixel-shuffle up-sampling [41] and then elementwise added (denoted by $\oplus$) with the features of adjacent layers. Finally, the more fine-grained feature on a small scale can be obtained.

Many existing methods [11,12,16,35,36] carried out multiple complex convolution operations at each downsampling layer to prevent the network from losing important detailed information, which will overload the network with large parameters. In our work, we only carry out a small number of feature extraction operations and then fuse multi-scale hierarchical information in small-scale space. This strategy can effectively reduce the computational cost.

Figure 3. The architecture of the multi-scale hierarchical information fusion scheme (MSHF). The purple block with down arrow represents the downsampling operation, the orange block with up arrow represents the upsampling operation.

The overall algorithm of the proposed MSHF is given in Algorithm 1.

Algorithm 1 Multi-scale Hierarchical fusion Algorithm

Input: f_0

Output: The output feature of MLHF f_{M_out}

1: For i in range (1, 5) do

2: $f_i = \text{downblock}(f_{i-1})$

3: $f_i = \text{resblock}(f_{i-1})$

4: End for

5: For i in range (3, 5) do

6: $f_{i-1} = \text{resblock}(f_{i-1} + \text{upsampling}(f_i))$

7: $f_{i-2} = \text{resblock}(f_{i-2} + \text{upsampling}(f_{i-1}))$

8: End for

9: $f_{M_out} = \text{resblock}(f_2)$

3.3. Multi-Feature Fusion Module Based on Attention Mechanism (MFFD)

3.3.1. Feature Distillation Normalization Block (FDBN)

After MSHF, a MFFD module is proposed to make the network continuously filter the useful channel feature information and eliminate the useless disturbance information. Generally, for reconstructing a sharp image well, it is necessary to increase the number of convolutional layers in the network so that the receptive field can be enlarged to get more information. However, this strategy is not a good choice in practice due to it dramatically increases the number of parameters, which will make the network converge slowly. In order to further reduce the network burden and pursue a lighter and faster network, it is extremely important to extract features through a distillation block. In our network, feature distillation normalization blocks (FDBN) are proposed for the first time in image deblurring and deraining method to extract useful features progressively. As shown in Figure 4, FDBN is mainly composed of the convolution layers with convolution kernel sizes as 1×1 and 3×3 , normalization block, and contrast-aware channel attention layer (CCA-Layer). Before each convolution layer, we perform channel segmentation on the input feature, which divides the feature into two parts. In the first part, the distilled feature $f_{c1,j}$ can be obtained by the 1×1 convolution. The 1×1 convolution aims to reduce the number of feature channels and the parameters. Then, we add a normalization layer [39,42] after 1×1 convolution. The normalization is designed to ensure the stability of network training and make the network converge more easily. In addition, since normalization can alleviate the dependence of a model on certain dataset, it also helps to improve the generalization ability of our network [43]. In the other part, through the 3×3 convolution, the coarse feature $f_{c3,j}$ is obtained. At the end of FDBN, the CCA-Layer is utilized to concatenate all the distilled features. Specifically, the operation of contrast allows us to obtain feature mappings of multiple spatial vectors. Through a series of FDBNs operations, the performance of our network can be steadily improved.

Figure 4. The architecture of the feature distillation normalization block (FDNB), Conv1 represents the convolution kernel sizes as 1×1 , Conv3 represents the convolution kernel sizes as 3×3 .

Overall, we can describe the process of FDNB more clearly with the following Equations:

$$f_{c1_1}, f_{c3_1} = F_s(\text{Conv1}(f_{in})), \quad (1)$$

$$f_{nc1_1} = \text{Normalization}(f_{c1_1}), \quad (2)$$

$$f_{c1_2}, f_{c3_2} = F_s(\text{Conv2}(f_{c3_1} + x)), \quad (3)$$

$$f_{nc1_2} = \text{Normalization}(f_{c1_2}), \quad (4)$$

$$f_{c1_3}, f_{c3_3} = F_s(\text{Conv3}(f_{c3_2} + x)), \quad (5)$$

$$f_{nc1_3} = \text{Normalization}(f_{c1_3}), \quad (6)$$

$$f_{c3} = \text{Conv4}(f_{c3_3} + x), \quad (7)$$

$$f_{out_i} = f_{in} + \text{CCA}(\text{Conv5}([f_{nc1_1}, f_{nc1_2}, f_{nc1_3}, f_{c3}])), \quad (8)$$

where f_{in} is the input feature of FDNB, F_s denotes the channel splitting operation, Conv_i represents i -th convolution layer, f_{ci_j} denotes the feature obtained by $i \times i$ Conv in j -th split operation, f_{nci_j} represents the feature obtained by $i \times i$ Conv in j -th normalization operation, x indicates residual. CCA indicates the operation of CCA-Layer, $[]$ represents the concatenation operation along the channel dimension, f_{out_i} is the output of i -th FDNB.

3.3.2. Fusion Mechanism

Unlike other networks [36–38] that simply stack several distillation modules to hierarchically extract features and only employ final features for the specific task, we rethink the distillation block in our proposed approach. We argue that although the multiple distillation modules (i.e., FDNBs) help the network to obtain richer image information with fewer parameters, the correlation between the intermedia features of each FDNB is ignored. At the same time, we also think about the problem of how to make the information of the last layer fully used. Thus, two different attention based feature fusion mechanisms are proposed to improve the representation ability of extracted features. One is the attention layer fusion module (ALFM) to learn the correlation between feature channels obtained by multiple FDNB layers. The other is the attention channel fusion module (ACFM) to describe the dependency between the inter-channel and intra-channel information in adjacent feature channels of the last layer.

The structure of ALFM is shown in Figure 5. Given the groups of feature channels obtained by N FDNB layers with the dimension of $N \times C \times W \times H$, we first reshape them along the channel and get an $NC \times WH$ feature matrix. Then, the feature matrix is transposed and multiplied by itself. In order to normalize the values of attention parameters, the softmax function is utilized in our ALFM. After softmax, we can get an attention matrix to reflect the correlation between channels. At last, we multiply the attention matrix by feature matrix with a scale factor θ and fuse it with the original feature channels by an addition operation. The process can be expressed in Equations (9) and (10):

$$m_{ij} = \text{softmax}(\text{reshape}(f_g) \otimes \text{reshape}(f_g)^T), \quad (9)$$

$$f_l = \theta \sum_{i=1}^N m_{ij} \otimes f_g + f_g, \quad (10)$$

where f_g is the input feature channels, $\otimes$ denotes the matrix multiplication, m_{ij} is the attention matrix, θ is initialized to 0 and it will be automatically optimized by the network. Here, it should be noted that our ALFM treats the feature channels of each FDNB layer separated rather than as a whole. Thus, the correlation between feature channels from both the same and different layers can be considered.

Figure 5. ALFM: The architecture of the attention layer fusion module.

The structure of ACFM is shown in Figure 6. The aim of ACFM is to model the interdependency between feature channels of the last FDNB layer by jointly considering channel and spatial information. Nevertheless, different from Niu et al. [35], who adopted 3D convolution to accomplish this task, our ACFM leverage a Pseudo-3D convolution strategy [44] to reduce the number of parameters. Taking the output of the last FDNB f_{out_n} as input, we first utilize two convolution kernels with the size of $1 \times 3 \times 3$ and $3 \times 1 \times 1$ to capture the spatial and channel correlations. Then, the attention matrix W obtained after the sigmoid function is element-wise multiplied by f_{out_n} . Finally, the weighted f_{out_n} is scaled by a factor α and fused with the original features.

$$f_c = f_{out_n} + \alpha \cdot \delta(\text{conv2}(\text{conv1}(f_{out_n}))) \odot f_{out_n}, \quad (11)$$

where $\delta(\cdot)$ is the sigmoid function, $\odot$ represents element-wise product, conv1 is $1 \times 3 \times 3$ convolution, conv2 is $3 \times 1 \times 1$ convolution, and α is an optimizable parameter which can be optimized by stochastic gradient descent based methods.

Through the attention mechanism in ALFM and ACFM, a more powerful feature representation can be achieved, which will compensate for the potential information loss in lightweight FDNB.

Overall, the loss function of our network can be expressed by Equation (12):

$$L_{MSE} = \frac{1}{wh} \sum_{x=1}^w \sum_{y=1}^h \left((I)_{x,y} - LFDN(I_{Blur/Rain})_{x,y} \right)^2, \quad (12)$$

where $I_{Blur/Rain}$ represents the input blurred or rainy image, $LFDN$ represents our network. I is the standard sharp image, w and h are the length and width of the input/output image, respectively.

Figure 6. ACFM: The architecture of the attention channel fusion module.

4. Experiments

4.1. Experimental Settings

We set the number of FDNB layers in our model as 4. The dimension of the feature maps in our fusion module is set as $N = 4$, $C = 50$, $W = 320$, $H = 180$. For model optimization, we adopt Adam with momentum as 0.9 and weight decay as 10^{-4} . The learning rate is initialized as 10^{-4} and decreases with a factor of 10 for every 5×10^5 iterations. All experiments were conducted using Pytorch on NVIDIA 2080Ti GPUs. Following other image deblurring and deraining works [19,22,45,46], PSNR, SSIM, model size, and inference time are adopted to evaluate our method. For comparative methods in the experiment, their results are directly quoted from their original papers.

4.1.1. Image Deblurring Dataset

Three benchmark datasets are employed for our image deblurring experiment.

GoPro dataset [9] consists of 3214 pairs of blurry and sharp images extracted from 33 sequences captured at 720×1280 resolution. The training and testing sets include 2103 and 1111 pairs, respectively.

Kohler dataset [47] consists of 4 images blurred with 12 different kernels for each of them, which is a standard benchmark dataset for evaluating the blind deblurring algorithms.

HIDE dataset [48] has 8422 sharp and blurry image pairs. The images are carefully selected from 31 high-fps videos that contain realistic outdoor scenes containing humans.

4.1.2. Image Deraining Dataset

In our deraining experiment, we use the following synthetic datasets.

The images of synthesizing Rain100L [49] and Rain100H [49] are selected from BSD200 [50]. Rain100L is a synthesized data set with only one type of rain streaks. Rain100H is a synthesized data set with five streak directions.

The Rain14000 [51] collects 1000 clean images from UCID dataset [52], BSD dataset [53], and Google image search to synthesize rainy images. Each clean image was used to generate 14 rainy images with different streak orientations and magnitudes.

The RainTest100 [27] consists of 100 images, where 50 images are randomly chosen from the last 500 images in UCID dataset, and 50 images are randomly chosen from the test set of BSD-500 [54] dataset.

In our deraining experiment, 11,200 images are randomly selected from Rain14000 [51] to form the training set. After training, the remaining 2800 images of Rain14000 [51], Rain100H [49], Rain100L [49], and RainTest100 [54] are used to evaluate the trained network.

In image deblurring experiment, we use GoPro [9] dataset that contains 2103 image pairs for training and 1111 pairs for evaluation. Furthermore, to demonstrate generalization performance of our method, we take our GoPro trained model and directly apply it to the test images of HIDE [48] and Kohler datasets [47]. The detailed information of benchmark datasets involved in our experiments can be seen in Table 1.

Table 1. Dataset description for image deblurring and deraining tasks.

Task	Deraining				Deblurring		
Datasets	Rain14000 [51]	Rain100L [49]	Rain100H [49]	RainTest100 [54]	GoPro [9]	HIDE [48]	Kolher [47]
Train samples	11,200	0	0	0	2103	0	0
Test samples	2800	100	100	100	1111	2025	64

4.2. Quantitative and Qualitative Evaluation on Deraining Task

Following the prior related work [19,22,24,26–28,55], we adopt the PSNR and SSIM metrics to evaluate the results of different methods for the image deraining task. It can be clearly seen from Table 2 that the proposed method outperforms other approaches on all four datasets. This indicates our LFDN can effectively remove the image degradation caused by the rain. Meanwhile, the experimental results in Table 2 also mean that our method is not dependent on a specific dataset and has a strong generalization ability to achieve good deraining results in various scenarios. Compared to the most recent MSPFN [19] method, the improvement of our method on Rain14000 [51] is very small, but the performance of our LFDN is much better than MSPFN on the other three benchmark datasets. PreNet [22] and MSPFN [19] also achieve good results on RAIN100L [49] and Rain14000 [51]. However, since there is no feature filter and distillation mechanism in them, their performance is inferior to ours. What is more, thanks to the design of ACFM and ALFM in the proposed LFDN, our method outperforms DerainNet [55]. DIDMDN [24] applied a small receptive field to capture small raindrops with small-scale features, while a large receptive field is employed to capture large raindrops with long scale features. However, due to the lack of fusion mechanism, its performance is worse than some other methods. UMRL [28] and RESCAN [27] use a multi-stage guidance model to generate sharp images, which also results in suboptimal network performance. In addition, in terms of computational complexity, our model runs at least two orders of magnitude faster.

Table 2. Image deraining results on different datasets.

Methods	Rain100H [49]		Rain100L [22]		Rain14000 [51]		Test100 [54]		Ave. Inf. Time (s)
	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	
DerainingNet [55]	14.92	0.592	27.03	0.884	24.31	0.961	22.77	0.810	-
SEMI [26]	16.56	0.486	25.03	0.842	24.43	0.782	22.35	0.788	4.567
DIDMDN [24]	17.35	0.524	25.23	0.741	28.13	0.867	22.56	0.818	2.789
UMRL [28]	26.01	0.832	29.18	0.923	29.97	0.923	24.41	0.829	2.552
RESCAN [27]	26.36	0.786	29.80	0.881	31.29	0.904	25.00	0.835	1.530
PreNet [22]	26.77	0.858	32.44	0.950	31.75	0.916	24.81	0.851	0.760
MSPFN [19]	28.66	0.860	32.40	0.933	32.82	0.930	27.50	0.876	0.230
LFDN (Ours)	29.12	0.893	32.79	0.961	32.90	0.935	28.44	0.880	0.043

From the qualitative comparison shown in Figure 7, it can be found that the image recovered by DIDMDN [24] is distorted in color, while the image details of UMRL [28] are still insufficient. However, our method can greatly restore the details blocked by the rain. The obvious improvement of our method over some recent works [19,22,24,26–28,55] may be attributed to the design of MSHF and MFFD.

Figure 7. Visual comparison on Rain14000 dataset.

4.3. Quantitative and Qualitative Evaluation on Deblurring Task

4.3.1. GoPro Dataset

We compare the deblurring performance of our model with some state-of-the-art methods [9–12,19,48] in terms of PSNR, SSIM, model size, and inference time on GoPro dataset. The quantitative results are shown in Table 3, and visual comparisons are shown in Figure 8. As shown in Table 3, our method owns a smaller model size (i.e., nearly 1.55 M parameters), which is 300X smaller than DeepDeblur [9]. DeepDeblur exploited a multi-scale CNN to restore sharp image, which may cause the heavy parameters. Through introducing the strategies of parameter sharing [10,11,13,20], GAN [15,16], hierarchical multi-patch [12,45], optical flow [21] and motion offsets [46], the number of parameters can be effectively reduced in other comparison methods. Among them, Gao et al. [11] got the smallest number of parameters because it employed a nested skip connection structure for the nonlinear transformation modules to replace stacked convolution layers or residual blocks. However, the authors of reference [11] only focus on the information of the last block and ignore a lot of semantic information contained in the intermedia layers. Different from [11], we utilize the ALFM to obtain the correlation between feature channels from the same and different layers and the ACFM is used to establish the interdependency between feature channels of the last FDNB layer. With the ALFM and ACFM as the fusion methods, the feature information of the middle layer and channels can be reused. Meanwhile, the fusion methods can reduce the weight of the network while ensuring the quality of image repair. Therefore, the PSNR and SSIM obtained by our LFDN are both superior to Gao et al. [11] and other approaches. For more details about the influence of network structure on the performance and model size of our LFDN, please refer to our ablation experiments in Section 4.4. In the visual comparison, it can be seen that most methods cannot recover the sharp object contour, cars, and persons from severe motion blur. For the second and forth images, we can still observe some noticeable blur artifacts in the results of some methods (such as [10–12]). Compared with other methods, our model can recover clearer details effectively.

Table 3. Performance and efficiency comparison on GoPro dataset.

Methods	PSNR	SSIM	Model Size (MB)	Ave. Inf. Time (s)
DeepDeblur [9]	29.08	0.841	303.6	15
Zhang et al. [13]	29.19	0.9306	37.1	1.4
Gao et al. [11]	30.92	0.9421	2.84	1.6
DeblurGAN [15]	28.70	0.927	37.1	0.85
Tao et al. [10]	30.10	0.9323	33.6	1.6
DeblurGANv2 [16]	29.55	0.934	15	0.35
DMPHN [12]	30.21	0.9345	21.7	0.03
SIS [20]	30.28	0.912	36.54	0.303
Yuan et al. [21]	29.81	0.9368	3.1	0.01
Tang et al. [45]	31.13	0.9507	31.1	0.088
Zhang et al. [46]	31.05	0.9485	26.3	-
LFDN (Ours)	31.60	0.932	1.55	0.029

Figure 8. Visual comparison on GoPro dataset.

4.3.2. Kohler Dataset

We report the quantitative results on Kohler dataset in Table 4. Visual comparisons are shown in Figure 9. From these results, we could see that our LFDN achieves the best quantitative (PSNR and SSIM) and qualitative performance. Furthermore, please be reminded that similarly to the GoPro case, the model size and average inference time of our method is still less than others.

Table 4. Performance and efficiency comparison on Kohler dataset.

Methods	PSNR	SSIM	Model Size (MB)	Ave. Inf. Time (s)
DeepDeblur [9]	26.48	0.807	303.6	15
Tao et al. [10]	26.57	0.8373	33.6	1.6
Zhang et al. [13]	24.21	0.7562	37.1	1.4
DeblurGAN [15]	26.10	0.807	37.1	0.85
DeblurGANv2 [16]	26.97	0.830	15	0.35
Cai et al. [56]	28.92	0.893	-	1200
Xu et al. [1]	27.47	0.811	-	13.41
LFDN (Ours)	30.98	0.9032	1.55	0.029

Figure 9. Visual comparison on Kohler dataset.

4.3.3. HIDE Dataset

To verify the validity of our method, we further evaluate our approach on HIDE testing set [48]. From the experimental results in Table 5 and Figure 10, we can clearly find that the performance of our method is better than others and the deblurred images obtained by our LFDN contains more detailed information.

Table 5. Performance comparison on HIDE dataset.

Methods	Sun et al. [57]	DeepDeblur [9]	Tao et al. [10]	Kupyn et al. [15]	Suin et al. [23]	DMPHN [12]	LFDN (Ours)
PSNR	23.21	27.43	28.60	26.44	29.98	29.09	30.07
SSIM	0.797	0.902	0.928	0.890	0.930	0.924	0.932
modle size (MB)	-	303.6	33.6	37.1	-	86.8	1.55
Ave. Inf. Time (s)	23.45	4.33	1.6	0.85	0.77	0.98	0.029

Figure 10. Visual comparison on HIDE dataset.

4.4. Ablation Study

In this section, we conduct several experiments to evaluate the effectiveness of each component in our method on both deblurring and deraining datasets: GoPro [9] and Rain100H [49]. First, we replace the multi-scale hierarchical fusion module (MSHF) with traditional convolution layers in our network. In this experiment, the stride of convolution in each layer is set as 2, and two residual models are utilized to resampling the feature. Then, the feature distillation normalization blocks (FDBNs) in the decoding stage of our approach are removed and the same number of traditional residual feature distillation blocks (RFDBs) or CNNs are employed for sharp image reconstruction. Finally, we discarded the attention based information fusion modules (i.e., ALFM and ACFM) and employed a standard concatenation to combine the features obtained by different FDBN layers and feature channels of the last FDBN. Through the above settings, we can get five new network structures, i.e., the proposed LFDN without MSHF, without FDBN (replaced by RFDB), without FDBN (replaced by CNN), without ALFM / ACFM and without all of the above components. In order to fairly compare the proposed model with different structures, we adopted the same parameter setting to train the networks. The results of ablation experiments on the GoPro [9] and Rain100H [49] datasets are summarized in Table 6.

Table 6. Ablation experimental results of our network on GroPro and Rain100H datasets.

MSHF	FDBN	ALFM/ACFM	PSNR on GroPro	SSIM on GroPro	PSNR on Rain100H	SSIM on Rain100H	Model Size (MB)
X	X	X	28.71	0.901	27.36	0.862	33.6
X	✓	✓	30.84	0.917	28.92	0.885	1.8
✓	X (RFDB)	✓	29.28	0.891	28.48	0.871	2.9
✓	X (CNN)	✓	28.56	0.882	28.04	0.867	4.3
✓	✓	X	31.01	0.921	28.98	0.886	2.4
✓	✓	✓	31.60	0.932	29.12	0.893	1.55

As can be seen from Table 6, without any of MSHF, FDBN, or ALFM/ACFM, the image recovery is not optimal. We can find that without MSHF, the value of PSNR decreases by 0.5, and the model size increases by 0.8. This means that MSHF can extract better features with a larger receiver field. Moreover, we can find that performing most of the data computation in the low-scale space is beneficial to reduce the network parameters. The feature fusion

in low-scale space is necessary to improve the accuracy. The effectiveness of the FDNB is the most obvious for the network performance. When the FDNB in our model is replaced by RFDB or CNN, its accuracy will be greatly decreased. This is because the RFDB has no ability to normalize the feature maps and CNN cannot filter out useless feature channel information. In order to make the network lightweight without loss of accuracy, we employ ALFM/ACFM mechanism to get finer-grained features. By the ablation experiments, we found that the value of PSNR and SSIM obtained by our model with ALFM/ACFM are higher than without them, which justifies the validity of the proposed ALFM/ACFM mechanism. In addition, we also provide a qualitative visual comparison in Figure 11.

Figure 11. Visual comparison on GoPro (first row) and Rain100H (second row) datasets. (a) without MSHF: Only without MSHF. (b) without FDNB (RFDB): Use traditional RFDB instead of the FDNB. (c) without FDNB (CNN): Use traditional CNN instead of the FDNB. (d) without ALFM/ACFM: Only without ALFM/ACFM. (e) LFDN: A complete network of ours.

In addition, the sensitivity of our LFDN to the parameters θ and α in ALFM and ACFM modules is tested. As we have mentioned in Section 3.3.2, these two parameters are automatically optimized by our network. Through experiments, we find that the optimal θ obtained by our network are -0.0015 and -0.0008 for deblurring and deraining tasks, while the optimal α obtained by our network are 0.1724 and 0.1137 for deblurring and deraining tasks. In order to justify these optimal parameter values, a comparative experiment is carried out to test the performance of our LFDN under a series of manually set parameter values. From the results in Tables 7 and 8, it can be found that the parameter values optimized by the network can achieve the best performance on both deblurring and deraining tasks.

Table 7. The performance comparison of different θ values.

θ	0.002	0.001	0	−0.001	−0.002	−0.0015	−0.0008
PSNR on GoPro	31.48	31.51	31.51	31.56	31.55	31.60	31.23
PSNR on Rain100H	28.95	28.97	29.01	29.10	29.07	29.01	29.12

Table 8. The performance comparison of different α values.

α	0.3	0.2	0.1	0	0.1724	0.1137
PSNR on GoPro	31.37	31.55	31.48	31.25	31.60	31.53
PSNR on Rain100H	28.44	29.02	29.09	28.99	29.03	29.12

Finally, we compare the performances of our LFDN with different numbers of FDNBs. From Table 9, we can find our method performs worse when the number of FDNB layers is small, while too many FDNBs would not further increase its performance. The results also justify that setting the number of FDNB layers as 4 is reasonable in our model.

Table 9. Performance comparison of our method under different numbers of distillation blocks.

Distillation Blocks Setting	1	2	3	4	5	6
PSNR on GroPro	29.82	29.73	30.77	31.60	31.52	31.61
PSNR on Rain100H	28.03	28.14	28.88	29.12	29.11	29.13
Model Size (MB)	1.46	1.49	1.52	1.55	1.58	1.61

5. Conclusions

In this work, we propose a lightweight fusion distillation network (LFDN) for image deblurring and deraining tasks. In order to make the network have fewer parameters and faster speed, the multi-scale hierarchical fusion module (MSHF) and feature distillation normalization block (FDNB) are adopted in the encoding and decoding stages, respectively. Moreover, two attention based modules are also proposed to improve the feature representation power in our approach. Through a large number of experiments on several benchmark datasets, the following points can be found. Firstly, the MSHF can extract and fuse the image feature in multiple small-scale spaces, which effectively eliminate redundant parameters while maintaining the rich image information. Secondly, the FDNB could help to improve the convergence speed and generalization ability of our proposed model. Thirdly, the attention based ALFM and ACFM modules also contribute to enhancing the performance of our model. Finally, the experimental results also demonstrate that our LFDN outperforms some state-of-the-art methods.

In our future work, we will extend the proposed method to some other image restoration tasks (such as image dehazing and denoising) and compare its performance with related approaches. Furthermore, we will also use other image quality assessment (IQA) methods to evaluate the performance of our method.

Author Contributions: Conceptualization, Y.Z., Y.L. and J.W.; methodology, Q.L. and H.X.; validation, Y.Z., Y.L. and J.W.; formal analysis, M.Q. and H.S.; data curation, Y.L., Y.Z. and J.W.; writing—original draft preparation, Y.Z.; writing—review and editing, H.X.; supervision, J.K.; project administration, J.K.; funding acquisition, J.K. and H.S. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported in part by National Key R&D Program of China (No. 2020YFA0714102), the NSFC (NO. 61907007), the Fund of the Jilin Provincial Science and Technology Department (No. 20200201199JC, 20200401081GX, 20200401086GX), and Fundamental Research Funds for the Central Universities (NO. 2412020FZ031, 2412020FZ029, 2412019FZ049).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The code of LFDN can be available at <https://github.com/LiQiang0307/LFDN> (accessed on 5 August 2021).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Xu, L.; Zheng, S.; Jia, J. Unnatural L0 Sparse Representation for Natural Image Deblurring. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Portland, OR, USA, 23–28 June 2013.
2. Li, L.; Pan, J.; Lai, W.; Gao, C.; Sang, N.; Yang, M. Blind Image Deblurring via Deep Discriminative Priors. *Int. J. Comput. Vis.* **2019**, *127*, 1025–1043. [\[CrossRef\]](#)
3. Chen, L.; Fang, F.; Wang, T.; Zhang, G. Blind Image Deblurring With Local Maximum Gradient Prior. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2019), Long Beach, CA, USA, 16–20 June 2019; Computer Vision Foundation/IEEE: Washington, DC, USA, 2019; pp. 1742–1750.
4. Luo, Y.; Xu, Y.; Ji, H. Removing Rain from a Single Image via Discriminative Sparse Coding. In Proceedings of the 2015 IEEE International Conference on Computer Vision (ICCV 2015), Santiago, Chile, 7–13 December 2015; IEEE Computer Society: Washington, DC, USA, 2015; pp. 3397–3405.

5. Chang, Y.; Yan, L.; Zhong, S. Transformed Low-Rank Model for Line Pattern Noise Removal. In Proceedings of the IEEE International Conference on Computer Vision (ICCV 2017), Venice, Italy, 22–29 October 2017; IEEE Computer Society: Washington, DC, USA, 2017; pp. 1735–1743.
6. Li, Y.; Tan, R.T.; Guo, X.; Lu, J.; Brown, M.S. Rain Streak Removal Using Layer Priors. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2016), Las Vegas, NV, USA, 27–30 June 2016; IEEE Computer Society: Washington, DC, USA, 2016; pp. 2736–2744.
7. Zhu, Z.; Wei, H.; Hu, G.; Li, Y.; Qi, G.; Mazur, N. A Novel Fast Single Image Dehazing Algorithm Based on Artificial Multiexposure Image Fusion. *IEEE Trans. Instrum. Meas.* **2021**, *70*, 1–23.
8. Zheng, M.; Qi, G.; Zhu, Z.; Li, Y.; Liu, Y. Image Dehazing by An Artificial Image Fusion Method based on Adaptive Structure Decomposition. *IEEE Sens. J.* **2021**, *70*, 1–23. [[CrossRef](#)]
9. Nah, S.; Kim, T.H.; Lee, K.M. Deep Multi-scale Convolutional Neural Network for Dynamic Scene Deblurring. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017.
10. Tao, X.; Gao, H.; Wang, Y.; Shen, X.; Wang, J.; Jia, J. Scale-recurrent Network for Deep Image Deblurring. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018.
11. Gao, H.; Tao, X.; Shen, X.; Jia, J. Dynamic Scene Deblurring With Parameter Selective Sharing and Nested Skip Connections. In Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 16–20 June 2019; pp. 3843–3851.
12. Zhang, H.; Dai, Y.; Li, H.; Koniusz, P. Deep Stacked Hierarchical Multi-Patch Network for Image Deblurring. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2019), Long Beach, CA, USA, 16–20 June 2019; Computer Vision Foundation/IEEE: Washington, DC, USA, 2019; pp. 5978–5986.
13. Zhang, J.; Pan, J.; Ren, J.; Song, Y.; Yang, M.H. Dynamic Scene Deblurring Using Spatially Variant Recurrent Neural Networks. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, USA, 18–22 June 2018.
14. Xu, L.; Ren, J.S.J.; Liu, C.; Jia, J. Deep Convolutional Neural Network for Image Deconvolution. In Proceedings of the Annual Conference on Neural Information Processing Systems 2014, Montreal, QC, Canada, 8–13 December 2014; pp. 1790–1798.
15. Kupyn, O.; Budzan, V.; Mykhailych, M.; Mishkin, D.; Matas, J. DeblurGAN: Blind Motion Deblurring Using Conditional Adversarial Networks. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, USA, 18–22 June 2018.
16. Kupyn, O.; Martyniuk, T.; Wu, J.; Wang, Z. DeblurGAN-v2: Deblurring (Orders-of-Magnitude) Faster and Better. *arXiv* **2019**, arXiv:1908.03826.
17. Gong, G.; Zhang, K. Local Blurred Natural Image Restoration Based on Self-Reference Deblurring Generative Adversarial Networks. In Proceedings of the 2019 IEEE International Conference on Signal and Image Processing Applications (ICSIPA 2019), Kuala Lumpur, Malaysia, 17–19 September 2019; IEEE: Washington, DC, USA, 2019; pp. 231–235.
18. Deng, S.; Wei, M.; Wang, J.; Liang, L.; Xie, H.; Wang, M. DRD-Net: Detail-recovery Image Deraining via Context Aggregation Networks. *arXiv* **2019**, arXiv:abs/1908.10267.
19. Jiang, K.; Wang, Z.; Yi, P.; Chen, C.; Huang, B.; Luo, Y.; Ma, J.; Jiang, J. Multi-Scale Progressive Fusion Network for Single Image Deraining. In Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR 2020), Seattle, WA, USA, 13–19 June 2020; IEEE: Washington, DC, USA, 2020; pp. 8343–8352.
20. Liu, Y.; Luo, Y.; Huang, W.; Qiao, Y.; Li, J.; Xu, D.; Luo, D. Semantic Information Supplementary Pyramid Network for Dynamic Scene Deblurring. *IEEE Access* **2020**, *8*, 188587–188599. [[CrossRef](#)]
21. Yuan, Y.; Su, W.; Ma, D. Efficient Dynamic Scene Deblurring Using Spatially Variant Deconvolution Network With Optical Flow Guided Training. In Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR 2020), Seattle, WA, USA, 13–19 June 2020; IEEE: Washington, DC, USA, 2020; pp. 3552–3561.
22. Ren, D.; Zuo, W.; Hu, Q.; Zhu, P.; Meng, D. Progressive Image Deraining Networks: A Better and Simpler Baseline. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2019), Long Beach, CA, USA, 16–20 June 2019; Computer Vision Foundation/IEEE: Washington, DC, USA, 2019; pp. 3937–3946.
23. Suin, M.; Purohit, K.; Rajagopalan, A.N. Spatially-Attentive Patch-Hierarchical Network for Adaptive Motion Deblurring. In Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR 2020), Seattle, WA, USA, 13–19 June 2020; IEEE: Washington, DC, USA, 2020; pp. 3603–3612.
24. Zhang, H.; Patel, V.M. Density-Aware Single Image De-Raining Using a Multi-Stream Dense Network. In Proceedings of the 2018 IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2018), Salt Lake City, UT, USA, 18–22 June 2018; IEEE Computer Society: Washington, DC, USA, 2018; pp. 695–704.
25. Zamir, S.W.; Arora, A.; Khan, S.; Hayat, M.; Khan, F.S.; Yang, M.; Shao, L. Multi-Stage Progressive Image Restoration. *arXiv* **2021**, arXiv:abs/2102.02808.
26. Wei, W.; Meng, D.; Zhao, Q.; Xu, Z.; Wu, Y. Semi-Supervised Transfer Learning for Image Rain Removal. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2019), Long Beach, CA, USA, 16–20 June 2019; Computer Vision Foundation/IEEE: Washington, DC, USA, 2019; pp. 3877–3886.

27. Li, X.; Wu, J.; Lin, Z.; Liu, H.; Zha, H. Recurrent Squeeze-and-Excitation Context Aggregation Net for Single Image Deraining. In *Computer Vision—ECCV 2018, Proceedings of the 15th European Conference, Munich, Germany, 8–14 September 2018*; Ferrari, V., Hebert, M., Sminchisescu, C., Weiss, Y., Eds.; Proceedings, Part VII; Lecture Notes in Computer Science; Springer: Berlin/Heidelberg, Germany, 2018; Volume 11211, pp. 262–277.
28. Yasarla, R.; Patel, V.M. Uncertainty Guided Multi-Scale Residual Learning-Using a Cycle Spinning CNN for Single Image De-Raining. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2019)*, Long Beach, CA, USA, 16–20 June 2019; Computer Vision Foundation/IEEE: Washington, DC, USA, 2019; pp. 8405–8414.
29. Purohit, K.; Rajagopalan, A.N. Region-Adaptive Dense Network for Efficient Motion Deblurring. In *Proceedings of the Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, Thirty-Second Innovative Applications of Artificial Intelligence Conference (IAAI 2020), Tenth AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI 2020)*, New York, NY, USA, 7–12 February 2020; AAAI Press: Menlo Park, CA, USA, 2020; pp. 11882–11889.
30. Chen, L.; Zhang, H.; Xiao, J.; Nie, L.; Shao, J.; Liu, W.; Chua, T. SCA-CNN: Spatial and Channel-Wise Attention in Convolutional Networks for Image Captioning. In *Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2017)*, Honolulu, HI, USA, 21–26 July 2017; IEEE Computer Society: Washington, DC, USA, 2017; pp. 6298–6306.
31. Lin, T.Y.; Dollar, P.; Girshick, R.; He, K.; Hariharan, B.; Belongie, S. Feature Pyramid Networks for Object Detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Honolulu, HI, USA, 21–26 July 2017; IEEE Computer Society: Washington, DC, USA, 2017.
32. Zhang, Y.; Kong, J.; Qi, M.; Liu, Y.; Wang, J.; Lu, Y. Object Detection Based on Multiple Information Fusion Net. *Appl. Sci.* **2020**, *10*, 418. [\[CrossRef\]](#)
33. Kim, J.; Choi, J.; Cheon, M.; Lee, J. RAM: Residual Attention Module for Single Image Super-Resolution. *arXiv* **2018**, arXiv:abs/1811.12043.
34. Khan, S.; Naseer, M.; Hayat, M.; Zamir, S.W.; Khan, F.S.; Shah, M. Transformers in Vision: A Survey. *arXiv* **2021**, arXiv:abs/2101.01169.
35. Niu, B.; Wen, W.; Ren, W.; Zhang, X.; Yang, L.; Wang, S.; Zhang, K.; Cao, X.; Shen, H. Single Image Super-Resolution via a Holistic Attention Network. In *Computer Vision—ECCV 2020, Proceedings of the 16th European Conference, Glasgow, UK, 23–28 August 2020*; Vedaldi, A., Bischof, H., Brox, T., Frahm, J., Eds.; Proceedings, Part XII; Lecture Notes in Computer Science; Springer: Berlin/Heidelberg, Germany, 2020; Volume 12357, pp. 191–207.
36. Hui, Z.; Wang, X.; Gao, X. Fast and Accurate Single Image Super-Resolution via Information Distillation Network. In *Proceedings of the 2018 IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2018)*, Salt Lake City, UT, USA, 18–22 June 2018; IEEE Computer Society: Washington, DC, USA, 2018; pp. 723–731.
37. Hui, Z.; Gao, X.; Yang, Y.; Wang, X. Lightweight Image Super-Resolution with Information Multi-distillation Network. In *Proceedings of the 27th ACM International Conference on Multimedia (MM 2019)*, Nice, France, 21–25 October 2019; Amsaleg, L., Huet, B., Larson, M.A., Gravier, G., Hung, H., Ngo, C., Ooi, W.T., Eds.; ACM: New York, NY, USA, 2019; pp. 2024–2032.
38. Liu, J.; Tang, J.; Wu, G. Residual Feature Distillation Network for Lightweight Image Super-Resolution. In *Computer Vision—ECCV 2020 Workshops, Proceedings of the European Conference on Computer Vision, Glasgow, UK, 23–28 August 2020*; Bartoli, A., Fusiello, A., Eds.; Proceedings, Part III; Lecture Notes in Computer Science; Springer: Berlin/Heidelberg, Germany, 2020; Volume 12537, pp. 41–55.
39. Karras, T.; Laine, S.; Aila, T. A Style-Based Generator Architecture for Generative Adversarial Networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2019)*, Long Beach, CA, USA, 16–20 June 2019; Computer Vision Foundation/IEEE: Washington, DC, USA, 2019; pp. 4401–4410.
40. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep Residual Learning for Image Recognition. In *Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2016)*, Las Vegas, NV, USA, 27–30 June 2016; IEEE Computer Society: Washington, DC, USA, 2016; pp. 770–778.
41. Shi, W.; Caballero, J.; Huszar, F.; Totz, J.; Aitken, A.P.; Bishop, R.; Rueckert, D.; Wang, Z. Real-Time Single Image and Video Super-Resolution Using an Efficient Sub-Pixel Convolutional Neural Network. In *Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2016)*, Las Vegas, NV, USA, 27–30 June 2016; IEEE Computer Society: Washington, DC, USA, 2016; pp. 1874–1883.
42. Huang, X.; Belongie, S.J. Arbitrary Style Transfer in Real-Time with Adaptive Instance Normalization. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV 2017)*, Venice, Italy, 22–29 October 2017; IEEE Computer Society: Washington, DC, USA, 2017; pp. 1510–1519.
43. Ioffe, S.; Szegedy, C. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. In *Proceedings of the 32nd International Conference on Machine Learning (ICML 2015)*, Lille, France, 6–11 July 2015; Volume 37, pp. 448–456.
44. Qiu, Z.; Yao, T.; Mei, T. Learning Spatio-Temporal Representation with Pseudo-3D Residual Networks. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV 2017)*, Venice, Italy, 22–29 October 2017; IEEE Computer Society: Washington, DC, USA, 2017; pp. 5534–5542.
45. Tang, K.; Xu, D.; Liu, H.; Zeng, Z. Context Module Based Multi-patch Hierarchical Network for Motion Deblurring. *Neural Process. Lett.* **2021**, *53*, 211–226. [\[CrossRef\]](#)

46. Zhang, Y.; Wang, C.; Maybank, S.J.; Tao, D. Self-supervised Exposure Trajectory Recovery for Dynamic Blur Estimation. *arXiv* **2020**, arXiv:abs/2010.02484.
47. Köhler, R.; Hirsch, M.; Mohler, B.J.; Schölkopf, B.; Harmeling, S. Recording and Playback of Camera Shake: Benchmarking Blind Deconvolution with a Real-World Database. In *Computer Vision—ECCV 2012, Proceedings of the 12th European Conference on Computer Vision, Florence, Italy, 7–13 October 2012*; Fitzgibbon, A.W., Lazebnik, S., Perona, P., Sato, Y., Schmid, C., Eds.; Proceedings, Part VII; Lecture Notes in Computer Science; Springer: Berlin/Heidelberg, Germany, 2012; Volume 7578, pp. 27–40.
48. Shen, Z.; Wang, W.; Lu, X.; Shen, J.; Shao, L. Human-Aware Motion Deblurring. In Proceedings of the ICCV2019, Seoul, Korea, 27–28 October 2019.
49. Yang, W.; Tan, R.T.; Feng, J.; Liu, J.; Guo, Z.; Yan, S. Deep Joint Rain Detection and Removal from a Single Image. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2017), Honolulu, HI, USA, 21–26 July 2017; IEEE Computer Society: Washington, DC, USA, 2017; pp. 1685–1694.
50. Martin, D.R.; Fowlkes, C.C.; Tal, D.; Malik, J. A Database of Human Segmented Natural Images and its Application to Evaluating Segmentation Algorithms and Measuring Ecological Statistics. In Proceedings of the Eighth International Conference on Computer Vision (ICCV-01), Vancouver, BC, Canada, 7–14 July 2001; IEEE Computer Society: Washington, DC, USA, 2001; Volume 2, pp. 416–425.
51. Fu, X.; Huang, J.; Zeng, D.; Huang, Y.; Ding, X.; Paisley, J.W. Removing Rain from Single Images via a Deep Detail Network. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2017), Honolulu, HI, USA, 21–26 July 2017; IEEE Computer Society: Washington, DC, USA, 2017; pp. 1715–1723.
52. Schaefer, G.; Stich, M. UCID: An uncompressed color image database. In Proceedings of the Storage and Retrieval Methods and Applications for Multimedia 2004, San Jose, CA, USA, 20 January 2004; Volume 5307, pp. 472–480.
53. Arbelaez, P.; Maire, M.; Fowlkes, C.C.; Malik, J. Contour Detection and Hierarchical Image Segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.* **2011**, *33*, 898–916. [[CrossRef](#)] [[PubMed](#)]
54. Zhang, H.; Sindagi, V.; Patel, V.M. Image De-raining Using a Conditional Generative Adversarial Network. *IEEE Trans. Circuits Syst. Video Technol.* **2017**, *30*, 3943–3956. [[CrossRef](#)]
55. Fu, X.; Huang, J.; Ding, X.; Liao, Y.; Paisley, J.W. Clearing the Skies: A Deep Network Architecture for Single-Image Rain Removal. *IEEE Trans. Image Process.* **2017**, *26*, 2944–2956. [[CrossRef](#)] [[PubMed](#)]
56. Cai, J.; Zuo, W.; Zhang, L. Dark and Bright Channel Prior Embedded Network for Dynamic Scene Deblurring. *IEEE Trans. Image Process.* **2020**, *29*, 6885–6897. [[CrossRef](#)]
57. Sun, J.; Cao, W.; Xu, Z.; Ponce, J. Learning a convolutional neural network for non-uniform motion blur removal. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2015), Boston, MA, USA, 7–12 June 2015; IEEE Computer Society: Washington, DC, USA, 2015; pp. 769–777.