

Noname manuscript No.
 (will be inserted by the editor)

Deep Image Deblurring: A Survey

Kaihao Zhang · Wenqi Ren · Wenhan Luo · Wei-Sheng Lai ·
 Björn Stenger · Ming-Hsuan Yang · Hongdong Li

Received: date / Accepted: date

Abstract Image deblurring is a classic problem in low-level computer vision, which aims to recover a sharp image from a blurred input image. Recent advances in deep learning have led to significant progress in solving this problem, and a large number of deblurring networks have been proposed. This paper presents a comprehensive and timely survey of recently published deep-learning based image deblurring approaches, aiming to serve the community as a useful literature review. We start by discussing common causes of image blur, introduce benchmark datasets and performance metrics, and summarize different problem formulations. Next we present a taxonomy of methods using convolutional neural networks (CNN) based on architecture, loss function, and application, offering a detailed review

Kaihao Zhang and Hongdong Li
 Australian National University, Australia
 E-mail: {kaihao.zhang, hongdong.li}@anu.edu.au

Wenqi Ren
 Institute of Information Engineering, Chinese Academy of Sciences, Beijing, China
 E-mail: renwenqi@iie.ac.cn

Wenhan Luo
 Tencent AI Lab, Shenzhen, China
 E-mail: whluo.china@gmail.com

Björn Stenger
 Rakuten Institute of Technology, Rakuten Group Inc.
 E-mail: bjorn@cantab.net

Weisheng Lai and Ming-Hsuan Yang
 School of Engineering, University of California at Merced, Merced, CA, USA
 E-mail: {wlai24, mhyang}@ucmerced.edu

and comparison. In addition, we discuss some domain-specific deblurring applications including face images, text, and stereo image pairs. We conclude by discussing key challenges and future research directions.

1 INTRODUCTION

Image deblurring is a classic task in low-level computer vision, which has attracted the attention from the image processing and computer vision community. The objective of image deblurring is to recover a sharp image from a blurred input image, where the blur can be caused by various factors such as lack of focus, camera shake, or fast target motion [1, 13, 49, 123]. Some examples are given in Figure 1.

Non-deep learning image deblurring methods often formulate the task as an inverse filtering problem, where a blurred image is modeled as the result of the convolution with blur kernels, either spatially invariant or spatially varying. Some early approaches assume that the blur kernel is known, and adopt classical image deconvolution algorithms such as Lucy-Richardson, or Wiener deconvolution, with or without Tikhonov regularization, to restore sharp images [105, 127, 142]. On the other hand, blind image deblurring methods assume the blur kernel is unknown and aim to simultaneously recover both the sharp image and the blur kernel itself. Since this task is ill-posed, various additional constraints are used to regularize the solution [6, 16, 24, 140]. While these non-deep learning methods show good performance in certain cases, they typically do not perform well under more complicated yet common scenarios such as images with strong motion blur.

Recent advances of deep learning techniques have revolutionized the field of computer vision; significant



Fig. 1 Examples of different blurry images. Images suffer from blur due to (a) camera shake, (b) out-of-focus scene, (c) moving object, and (d) multiple causes, respectively.

progress has been made in numerous domains, including image classification [34, 118] and object detection [33, 42, 99, 164]. Image deblurring is no exception: a large number of deep learning methods have been developed for single image and video deblurring, and have advanced the state of the art. However, the introduction of new methods with different network designs makes it challenging to obtain a rapid overview of the field. This paper aims to fill this gap by providing a survey of recent advances, and to serve a reference point for new researchers.

Specifically, we will focus the discussion on recently published deep learning based image and video deblurring methods. The aims of this paper are:

- To review the preliminaries for image deblurring, including problem definitions, causes of blur, deblurring approaches, quality assessment metrics, and benchmark datasets for performance evaluation.
- To discuss new developments of deep learning models for single image and video deblurring and provide a taxonomy for categorizing the existing methods (see Figure 2).
- To analyze the challenges of image deblurring and discuss research opportunities.

The paper is organized as follows. In Section 2, we discuss the problem formulation, the causes of blur,

the types of deblurring, and image quality metrics. Sections 3 and 4 introduce the CNN-based non-blind and blind image deblurring methods, respectively. Loss functions applied in deep deblurring methods are discussed in Section 5. We introduce public benchmark datasets and evaluations in Sections 6 and 7, respectively. In Section 8, we review three deblurring methods for specific domains, for face, text, and stereo images. Finally, we discuss the challenges and future opportunities in this research area.

2 Preliminaries

2.1 Problem Formulation

Image blur can be caused by various factors during image capture: camera shake, in-scene motion, or out-of-focus blur. We denote a blurred image I_b as

$$I_b = \Phi(I_s; \theta_\eta), \quad (1)$$

where Φ is the image blur function, and θ_η is a parameter vector. I_s is the latent sharp version of the blurred image I_b . Deblurring methods can be categorized into non-blind and blind methods, depending on whether or not the blur function is known (see Section ??). The goal of image deblurring is to recover a sharp image, *i.e.*, finding the inverse of the blur function, as

$$I_{db} = \Phi^{-1}(I_b; \theta_\nu), \quad (2)$$

where Φ^{-1} is the deblurring model, θ_ν represents its parameters, and I_{db} is the deblurred image, which is the estimate of the latent sharp image I_s . Take deep deblurring methods as an example, Φ^{-1} and θ_ν can be regarded as the network and its parameters, respectively.

Motion Blur. An image is captured by measuring photons over the time period of camera exposure. Under bright illumination the exposure time is sufficiently short for the image to capture an instantaneous moment. However, a longer exposure time may result in motion blur. Numerous methods directly model the degradation process as a convolution process by assuming that the blur is uniform across the entire image:

$$I_b = K * I_s + \theta_\mu, \quad (3)$$

where K is the blur kernel and θ_μ represents additive Gaussian noise. In such an image, any object moving with respect to the camera will look blurred along the direction of relative motion. When we use Eq.3 to represent the blur process Eq.1, θ_η corresponds to a blur kernel and Gaussian noise, while Φ corresponds to convolution and sum operator. For camera shake, motion

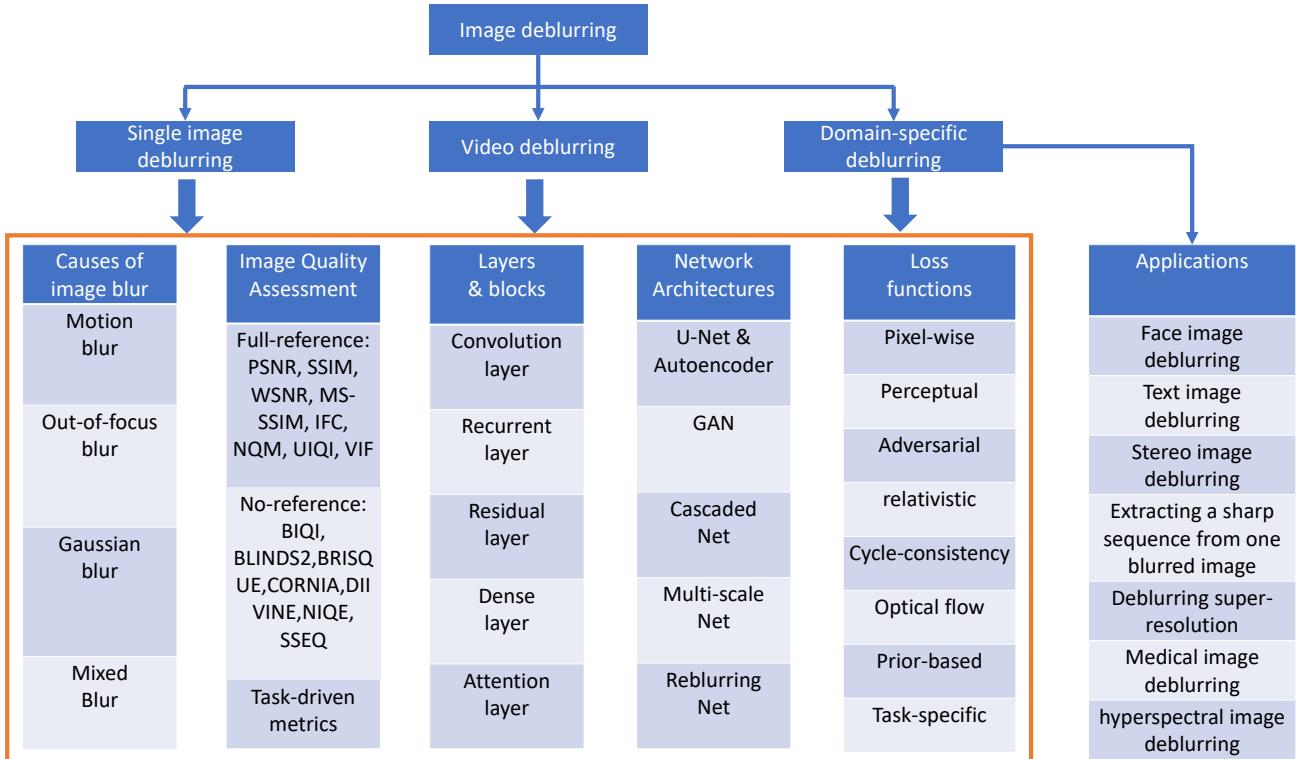


Fig. 2 Taxonomy of existing deep image deblurring techniques reviewed in this survey.

blur occurs in the static background, while, in the absence of camera shake, fast moving objects will cause these objects to be blurred while the background remains sharp. A blurred image can naturally contain blur caused by both factors. Early methods model blur using shift-invariant kernels [24, 141], while more recent studies address the case of non-uniform blur [26, 57, 58, 84, 129].

Out-of-focus Blur. Aside from motion blur, image sharpness is also affected by the distance between the scene and the camera's focal plane. When objects are in focus, they are exactly on the focal plane. When objects are at different distances to the camera, parts of the scene will be in focus, while others appear blurry. The Point Spread Function (PSF) for out-of-focus blur [71] is often modeled as:

$$K(x, y) = \begin{cases} \frac{1}{\pi r^2}, & \text{if } (x - k)^2 + (y - l)^2 \leq r^2, \\ 0, & \text{elsewhere,} \end{cases} \quad (4)$$

where (k, l) is the center of the PSF and r the radius of the blur. Out-of-focus deblurring has applications in saliency detection [43], defocus magnification [5] and image refocusing [157]. To address the problem of out-of-focus blur, classic methods remove blurry artifacts via blur detection [115] or coded apertures [75]. Deep

neural networks have been used to detect blur regions [128, 159] and predict depth [4] to guide the deblurring process.

Gaussian Blur. Gaussian convolution is a common simple blur model used in image processing, defined as

$$G(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}}, \quad (5)$$

where x and y are the distance from the origin in the horizontal and vertical axis, respectively, σ is the standard deviation. Several classic methods have been developed to remove the Gaussian blur [11, 39, 130].

Mixed Blur. In many real-world scenes multiple factors contribute to blur, such as camera shake, object motion, and depth variation. For example, when a fast-moving object is captured at an out-of-focus distance, the image may include both motion blur and out-of-focus blur as shown in Figure 1(d). To synthesize this type of blurry image, one option is to firstly transform sharp images to their motion-blurred versions (*e.g.*, by averaging neighboring sharp frames taken in sequence) and then apply an out-of-focus blur kernel based on Eq. 4. Alternatively, one can train a blurring network to directly generate realistically blurred images [12, 152].

Other than the main types of blur illustrated above, there may be other causes such as channel-dependent blur resulting from chromatic aberration [119, 126].

2.2 Image Quality Assessment

Methods for image quality assessment (IQA) can be classified into subjective and objective metrics. Subjective approaches are based on human judgment, which may not require a reference image. One representative metric is the Mean Opinion Score (MOS) [37], where people rate the quality of images on a scale of 1-5. MOS values vary based on different opinions, and methods relying on these scores typically take the statistics of opinion scores into account. For image deblurring, most existing methods are evaluated on objective assessment scores, which can be further split into two categories: full-reference and no-reference IQA metrics.

Full-Reference Metrics. Full-reference metrics assess the image quality by comparing the restored image with the ground-truth (GT). Such metrics include PSNR [36], SSIM [134], WSNR [77], MS-SSIM [135], IFC [110], NQM [19], UIQI [133], VIF [109], and LPIPS [156]. Among these, PSNR and SSIM are the most commonly used metrics in image restoration tasks [26, 57, 58, 84, 114, 121, 129, 150, 152]. On the other hand, LPIPS and E-LPIPS are able to accurately predict the human judgment of image quality [51, 156].

No-Reference Metrics. While the full-reference metrics require a ground-truth image for evaluation, no-reference metrics use only the deblurred images to measure the quality. To evaluate the performance of deblurring methods on real-world images, several no-reference metrics have been used, such as BIQI [80], BLINDS2 [104], BRISQUE [78], CORNIA [147], DIIVINE [81], NIQE [79], and SSEQ [68]. Further, a number of metrics have been developed to evaluate the performance of image deblurring algorithms by comparing the effect on the accuracy of different vision tasks, such as object detection and recognition [64, 146].

3 Non-Blind Deblurring

The goal of image deblurring is to recover the latent image I_s from a given blurry one I_b . If the blur kernel is given, the problem is also known as non-blind deblurring. Even if the blur kernel is available, the task is challenging due to sensor noise and the loss of high-frequency information.

Some non-deep methods employ natural image priors, e.g., global [55] and local [166] image priors, either

in the spatial domain [100] or in the frequency domain [56] to reconstruct sharp images. To overcome undesired ring artifacts, Xu *et al.* [141] and Ren *et al.* [102] combine spatial deconvolution and deep neural networks. In addition, several approaches have been proposed to handle saturated regions [17, 136] and to remove unwanted artifacts caused by image noise [52, 87].

We summarize existing deep learning based non-blind methods in Table 1. These approaches can be broadly categorized into two groups: the first group uses deconvolution followed by denoising, while the second group directly employs deep networks.

Deconvolution with Denoising. Representative algorithms in this category include [102, 106, 141, 149, 154]. Schuler *et al.* [106] develop a multi-layer perceptron (MLP) to deconvolve images. This approach first recovers a sharp image through a regularized inverse in the Fourier domain, and then uses a neural network to remove artifacts which are produced in the deconvolution process. Xu *et al.* [141] use a deep CNN to deblur images containing outliers. This algorithm applies singular value decomposition to a blur kernel and draws a connection between traditional optimization-based schemes and CNNs. However, the model needs to be retrained for different blur kernels. Inspired by the low-rank property of the pseudo-inverse kernel in [141], Ren *et al.* [102] propose a generalized deep CNN to handle arbitrary blur kernels in a unified framework without re-training for each kernel. However, low-rank decompositions of blur kernels can lead to a drop in performance. Both methods [102, 141] concatenate a deconvolution CNN and a denoising CNN to remove blur and noise, respectively. However, such denoising networks are designed to remove additive white Gaussian noise and poorly handle outliers or saturated pixels in blurry images. In addition, these non-blind deblurring networks need to be trained for a fixed noise level to achieve good performance, which limits their use in general cases. Kruse *et al.* [56] propose a Fourier Deconvolution Network (FDN) by unrolling an iterative scheme, where each stage contains an FFT-based deconvolution module and a CNN-based denoiser. Data with multiple noise levels is synthesized for training to achieve better deblurring and denoising performance.

The above methods tend to learn denoise modules for non-blind image deblurring. Learning denoising modules is one type of method for learning priors, which will be discussed in the following.

Learning priors for deconvolution. Bigdeli *et al.* [7] learn a mean-shift vector field representing a smoothed version of the natural image distribution, and use gradient descent to minimize the Bayes risk for non-blind

Table 1 Overview of deep single image non-blind deblurring methods, where ‘‘Convolution’’ denotes convolving sharp images with blur kernels using Eq. 3 to synthesize training data.

Method	Category	Blur type	Dataset	Architecture	Key idea
DCNN [141]	Uniform	Gaussian, disk	Convolution	CNN	first work to combine traditional optimization-based schemes and neural networks
IRCNN [154]		Gaussian, motion			learn set of CNN denoisers, use as modular part of model-based optimization methods to tackle inverse problems
FCNN [149]		Motion			adaptively learn image priors to preserve image details and structure with a robust L1 loss
FDN [56]		Motion			learn a CNN-based prior with an FFT-based deconvolution scheme
GLRA [102]		Gaussian, disk, motion			use generalized low-rank approximations of blur kernels to initialize the CNN parameters
DUBLID [65]		Motion			recast a generalized TV-regularized algorithm into a deep network for blind image deblurring
RGDN [30]		Motion			incorporate deep neural networks into a fully parameterized gradient descent scheme
DWDN [21]		Motion, Gaussian			apply explicit deconvolution in feature space by integrating a classical Wiener deconvolution framework

deblurring. Jin *et al.* [46] use a Bayesian estimator for simultaneously estimating the noise level and remove blur. They also propose a network (GradNet) to speed up the deblurring process. In contrast to learning a fixed image prior, GradNet can be integrated with different priors and improves existing MAP-based deblurring algorithms. Zhang *et al.* [154] train a set of discriminative denoisers and integrate them into a model-based optimization framework to solve the non-blind deblurring problem.

The learned priors from [7, 46, 154] are not always able to handle blurry images with outliers. Without outlier handling, these non-blind deblurring approaches tend to generate ringing artifacts, even in cases when the estimated kernel is accurate. We note that some super-resolution methods (with a scale factor of 1) can be adopted for the task of non-blind deblurring due to their similar problem formulation, *e.g.*, USRNet [153].

4 Blind Deblurring

In this section, we discuss recent blind deblurring methods. For blind deblurring, both the latent image and the blur kernel are unknown. Early blind deblurring methods focus on removing uniform blur [16, 24, 76, 107, 140]. However, real-world images typically contain non-uniform blur, where different regions in the same image are generated by different blur kernels [103]. Numerous approaches have been developed to address non-uniform blur by modeling the blur kernel from 3D camera motion [35, 137]. Although these approaches can model out-of-plane camera shake, they cannot han-

dle dynamic scenes, which motivated the use of blur fields of moving objects [10, 27, 40]. Motion discontinuities and occlusions make the accurate estimation of blur kernels challenging. Recently, several deep learning based methods have been proposed for the deblurring of dynamic scenes [26, 84, 129].

Tables 2 and 3 summarize representative single image and video deblurring methods, respectively. To analyse these methods, we first introduce frame aggregation methods for network input. We then review the basic layers and blocks used in existing deblurring networks. Finally, we discuss the architectures, as well as the advantages and limitations of current methods.

4.1 Network inputs and frame aggregation

Single image deblurring networks take a single blurry image as input and generate the corresponding deblurred result. Video deblurring methods take multiple frames as input and aggregate the frame information in either the image or feature domain. Image-level aggregation algorithms, *e.g.*, Su *et al.* [120], stack multiple frames as input and estimate the deblurred result for the central frame. On the other hand, feature-level aggregation approaches, *e.g.*, Zhou *et al.* [162] and Kim *et al.* [41], first extract features from the input frames and then fuse the features for predicting the deblurred results.

4.2 Basic Layers and Blocks

This section briefly reviews the most common network layers and blocks used for image deblurring.

Convolutional layer. Numerous methods [9, 50, 107, 123] train 2D CNNs to directly recover sharp images without kernel estimation steps [3, 29, 70, 73, 82, 89, 148, 114, 155]. On the other hand, several approaches use additional prior information, such as depth [63] or semantic labels [112], to guide the deblurring process. In addition, 2D convolutions are also adopted by all video deblurring methods [2, 41, 53, 83, 85, 116, 120, 132, 138]. The main difference between single image and video deblurring are 3D convolutions, which can extract features from both spatial and temporal domains [151].

Recurrent layer. For single image deblurring, recurrent layers can extract features across images at multiple scales in a coarse-to-fine scheme. Two representative methods are SRN [129] and PSS-SRN [26]. SRN builds a coarse-to-fine architecture to remove motion blur via a shared-weight deep autoencoder, while PSS-SRN includes a selective parameter sharing scheme, which leads to improved performance over SRN.

Recurrent layers can also be used to extract temporal information from neighboring frames in videos [41, 72, 85, 94, 161, 162]. The main difference from single image deblurring is that the recurrent layers in video-based methods extract features from neighboring images, rather than transferring information across only one input image at different scales. These methods can be categorized into two groups. The first group transfers the feature maps from the last step to the current network to obtain finer deblurred frames [41, 85]. The second type of methods generates sharp frames via directly inputting deblurred frames from the last step [162].

Residual layer. To avoid vanishing or exploding gradients during training, the global residual layers are utilized to directly connect low-level and high-level layers in the area of image deblurring [57, 84, 158]. One representative method using this architecture is DeblurGAN [57] where the output of the deblurring network is added to the input image to estimate the sharp image. In this case, the architecture is equivalent to learning global residuals between blurry and sharp images. The local ResBlock uses local residual layers, similar to the residual layers in ResNet [34], and these are widely used in image deblurring networks [26, 41, 57, 84, 89, 129]. Both kinds of residual layers, local and global, are often combined to achieve better performance.

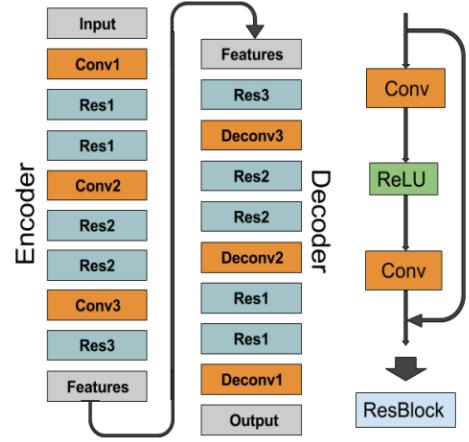


Fig. 3 Deep single image deblurring network based on the Deep Auto-Encoder (DAE) architecture [89].

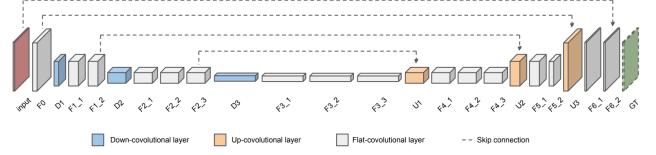


Fig. 4 Deep video deblurring network based on the DAE architecture [120].

Dense layer. Using dense connections can facilitate addressing the gradient vanishing problem, improving feature propagation, and reducing the number of parameters. Purohit *et al.* [96] propose a region-adaptive dense network composed of region adaptive modules to learn the spatially varying shifts in a blurry image. These region adaptive modules are incorporated into a densely connected auto-encoder architecture. Zhang *et al.* [152] and Gao *et al.* [26] also apply dense layers to build their deblurring networks in which DenseBlocks are used to replace the CNN layers or ResBlocks.

Attention layer. The attention layer can help deep networks focus on the most important image regions for deblurring. Shen *et al.* [114] propose an attention-based deep deblurring method consisting of three separate branches to remove blur from the foreground, the background, and globally, respectively. Since image regions containing people are often of the most interest, the attention module detects the location of people to deblur images using the guidance of a human-aware map. Other methods employ attention to extract better feature maps, *e.g.*, using self-attention to generate a non-locally enhanced feature map [96].

Table 2 Overview of deep single image blind deblurring methods. In the ‘Dataset’ column ‘averaging’ refers to averaging over temporally consecutive sharp frames to synthesize training data.

Method	Category	Blur type	Dataset	Architecture	Key idea
Learning-to-Deblur [107]	Uniform	Motion	Convolution	Cascade	The first stage uses a CNN to estimate blur kernels and latent images. The second stage operates on the blurry images and latent image for kernel estimation.
TextDBN [38]		Motion & defocus		CNN	Trains a CNN for blind deblurring and denoising.
SelfDeblur [98]		Gaussian & motion		DAE	Two generative networks capture the blur kernel and a latent sharp image, respectively, which is trained on blurry images.
MRFCNN [123]	Non-uniform		Convolution	CNN	Estimate motion kernels from local patches via CNN. An MRF model predicts the motion blur field.
NDEBLUR [9]			Convolution	CNN	Train a network to generate the complex Fourier coefficients of a deconvolution filter, which is applied to the input patch.
MSCNN [84]			Averaging	MS-CNN	A multi-scale CNN generates a low-resolution deblurred image and a deblurred version at the original resolution.
BIDN [89]			Convolution	DAE	The network regresses over encoder-features to obtain a blur invariant representation, which is fed into a decoder to generate the sharp image.
MBKEN [144]			Convolution	Cascade	A two-stage CNN extracts sharp edges from blurry images for kernel estimation.
RNN_Deblur [150]			Convolution	RNN	Deblurring via a spatially variant RNN, whose weights are learned via a CNN.
SRN [129]			Averaging	MS-LSTM	Deblurring via a scale-recurrent network that shares network weights across scales.
DeblurGAN [57]		Motion	Averaging	GAN	A conditional GAN-based network generates realistic deblurred images.
UCSDBN [73]			Convolution	Cycle-GAN	An unsupervised GAN performs class-specific deblurring using unpaired images as training data.
DMPHN [148]			Convolution	DAE	A DAE network recovers sharp images based on different patches.
DeepGyro CNN [82]			Convolution	DAE	A motion deblurring CNN makes use of the camera’s gyroscope readings.
PSS_SRN [26]			Averaging	MS-LSTM	A selective parameter sharing scheme is applied to the SRN architecture and ResBlocks are replaced by nested skip connections.
DR_UCSDBN [70]			Convolution	Cycle-GAN	Unsupervised domain-specific deblurring method by disentangling the content and blur features from input images.
Dr-Net [3]			Averaging	CNN	A network to learn both the image prior and data fidelity terms via Douglas-Rachford iterations.
DeblurGAN-v2 [58]			Averaging	GAN	An extension of DeblurGAN using a feature pyramid network and wide range of backbone networks for better speed and accuracy.
RADN [96]			Averaging	DAE	Region-adaptive dense deformable module to discover spatially varying shifts.
DBRBGAN [152]			Averaging	Reblur	Two networks, BGAN and DBGAN, which learn to blur and to deblur, respectively.
SAPHN [121]			Averaging	DAE	Content-adaptive architecture to remove spatially-varying image blur.
ASNet [50]			Convolution	DAE	DAE framework, which first estimates the blur kernel in order to recover sharp images.
EBMD [44]			Averaging	DAE	An event-based motion deblurring network, introducing a new dataset, DAVIS240C.

Table 3 Overview of deep video deblurring methods.

Method	Category	Blur type	Dataset	Architecture	Key idea
DeBlurNet [120]				DAE	Five neighboring blurry images are stacked and fed into a DAE to recover the center sharp image.
STRCNN [41]				RNN-DAE	Recurrent architecture which includes a dynamic temporal blending mechanism for information propagation.
PICNN [2]				DAE	Permutation invariant CNN which consists of several U-Nets taking a sequence of burst images as input.
DBLRCGAN [151]				GAN	GAN-based video deblurring method, using a 3D CNN to extract spatio-temporal information.
Reblur2deblur [12]	Non-uniform	Motion	Averaging	Reblur	Three consecutive blurry images are fed into the reblur2deblur framework to recover sharp images, which are used to compute the optical flow and estimate a blur kernel for reconstructing the input.
IFIRNN [85]				RNN-DAE	RNN-based video deblurring network, where a hidden state is transferred from past frames to the current frame.
EDVR [132]				DAE	Pyramid, Cascading and Deformable (PCD) module for frame alignment and a Temporal and Spatial Attention (TSA) fusion module, followed by a reconstruction module to restore sharp videos.
STFAN [162]				DAE	STFAN takes the current blurred frame, the preceding blurred and restored frames as input and recovers a sharp version of the current frame.
CDVD-TSP [90]				Cascade	Cascaded deep video deblurring which first calculates the sharpness prior, then feeds both the blurry images and the prior into an DAE.

4.3 Network Architectures

We categorize the most widely used network architectures for image deblurring into five sets: Deep auto-encoders (DAE), generative adversarial networks (GAN), cascaded networks, multi-scale networks, and reblurring networks. We discuss these methods in the following sections.

Deep auto-encoders (DAE). A deep auto-encoder first extracts image features and a decoder reconstructs the image from these features. For single image deblurring, many approaches use the U-Net architecture with a residual learning technique. [26, 89, 112, 116, 129]. In some cases additional networks help exploiting additional information for guiding the U-Net. For example, Shen *et al.* [112] propose a face parsing/segmentation network to predict face labels as priors, and use both, blurry images and predicted semantic labels, as the input to a U-Net. Other methods apply multiple U-Nets to obtain better performance. Tao *et al.* [129] analyze different U-Nets as well as DAE, and propose a scale-recurrent network to process blurry images. The first U-net obtains coarse deblurred images, which are fed into another U-Net to obtain the final result. The work in [113] combines the two ideas, using a deblurring network to obtain coarse deblurred images, then feeding

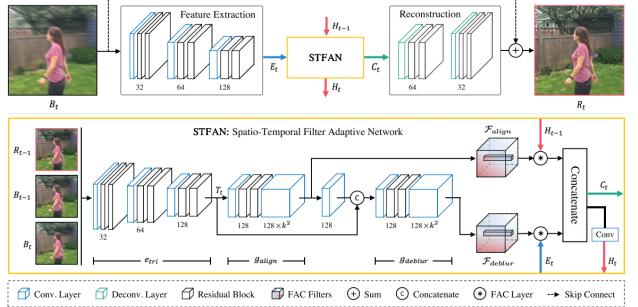


Fig. 5 Architecture to extract spatio-temporal information based on a STFAN model [162].

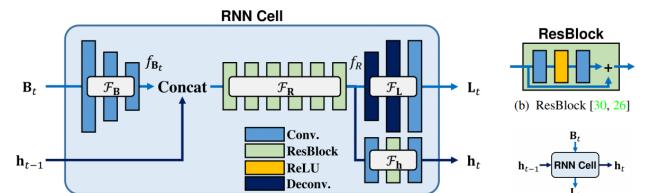


Fig. 6 Architecture to extract spatio-temporal information based on an RNN model [85].

them into a face parsing network to generate semantic labels. Finally, both, coarse deblurred images and labels, are fed into a U-Net to obtain the final deblurred images.

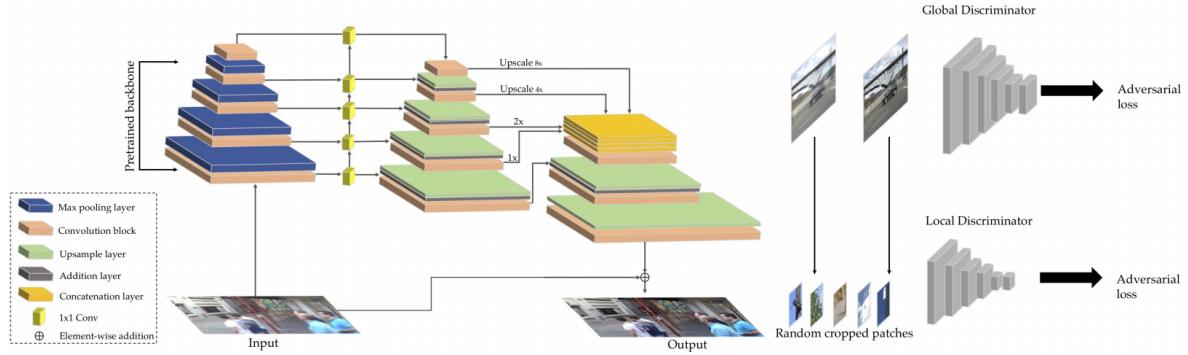


Fig. 7 Deep single image deblurring network based on the GAN architecture [58].

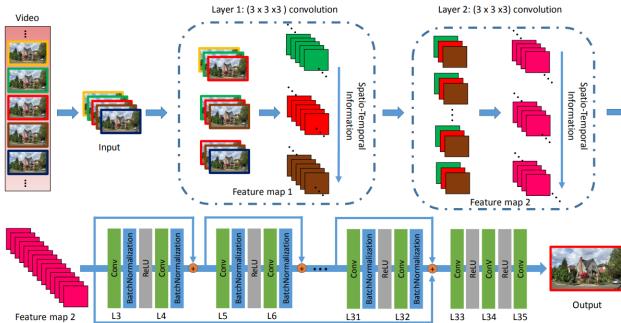


Fig. 8 Deep video deblurring network based on the GAN architecture [151].

Video deblurring methods can be split into two groups based on their input. The first group takes a stack of neighboring blurry frames as input to extract spatio-temporal information. Su *et al.* [120] and Wang *et al.* [132] design DAE architectures to remove blur from videos by feeding several consecutive frames into the encoder, and the decoder recovers the sharp central frame. Features extracted from different layers of the encoder are element-wise added to the corresponding decoder layers as shown in Figure 4, which accelerates convergence and generates sharper images. The second approach is to feed a single blurry frame into an encoder to extract features. Various modules have been developed to extract features from neighboring frames, which are jointly fed into the decoder to recover the deblurred frame [162]. The features from neighboring frames can also be fed into the encoder for feature extraction [41]. Numerous deep video deblurring algorithms based on this architecture have been developed [85, 116, 132]. The main difference is the module that models the temporal information from neighboring frames, *e.g.*, STFAN [162] in Figure 5 and an RNN module [85] in Figure 6.

Generative adversarial networks (GAN). GANs have been widely used in image deblurring [57, 58, 84,

112] in recent years. Most GAN-based deblurring models share the same strategy: the generator (Figure 7) generates sharp images such that the discriminator cannot distinguish them from real sharp images. Kupyn *et al.* [57] proposed DeblurGAN, an end-to-end conditional GAN for motion deblurring. The generator of DeblurGAN contains two-strided convolution blocks, nine residual blocks, and two transposed convolution blocks to transform a blurry image to its corresponding sharp version. This method is further extended to DeblurGAN-v2 [58], which adopts a relativistic conditional GAN and a double-scale discriminator, which consists of local and global branches as in [42]. The core block of the generator is a feature pyramid network, which improves efficiency and performance. An adversarial loss is employed by Nah *et al.* [84] and Shen *et al.* [112] to generate better deblurred images.

GANs are also used in video deblurring networks. The main difference from single image deblurring is the generator, which also considers temporal information from neighboring frames. A 3DCNN model was developed by Zhang *et al.* [151] to exploit both spatial and temporal information to restore sharp details using adversarial learning, see Figure 8. Kupyn *et al.* [58] proposed DeblurGAN-v2 to restore sharp videos via modifying the single image deblurring method of DeblurGAN.

Cascaded networks. A cascaded network contains several modules, which are sequentially concatenated to construct a deeper structure. Cascaded networks can be divided into two groups. The first one uses different architectures in each cascade. For example, Schuler *et al.* [107] propose a two-stage cascaded network, see Figure 9. The input to the first stage is a blurry image, and the deblurred output is fed into the second stage to predict blur kernels. The second group re-trains the same architecture in each cascade to generate deblurred images. Deblurred images from preceding stages are fed into the same type of network to generate finer de-

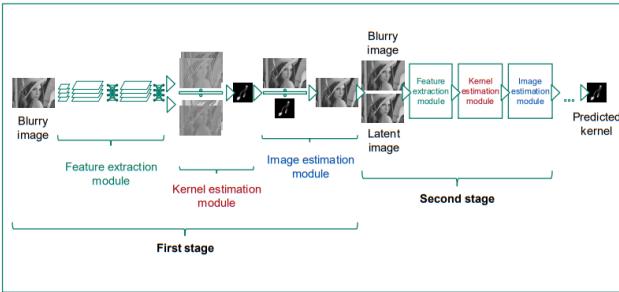


Fig. 9 Deep network for single image deblurring based on a cascaded architecture [107].

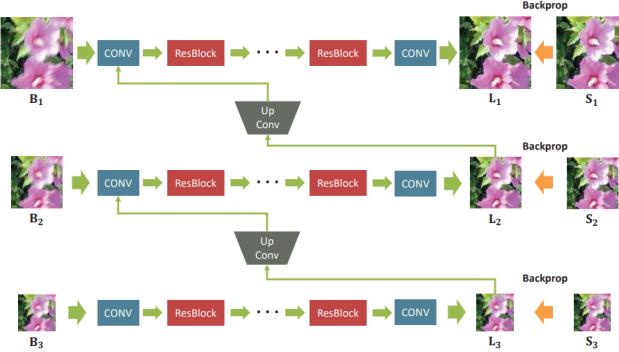


Fig. 10 Deep network for single image deblurring based on a multi-scale architecture [84].

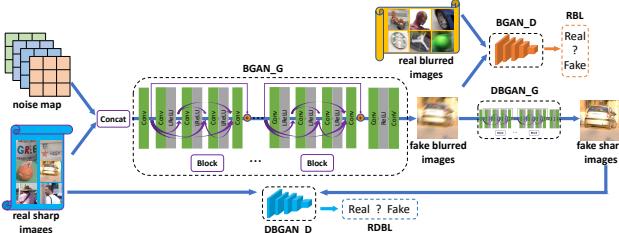


Fig. 11 Deep network for single image deblurring based on a reblurring architecture [152].

blurred results. This cascading scheme can be used in almost all deblurring networks. Although this strategy achieves better performance, the number of CNN parameters increases significantly. To reduce these, recent approaches share parameters in each cascade [90].

Multi-scale networks. Different scales of the input image describe complementary information [20, 22, 139]. The strategy of multi-scale deblurring networks is to first recover low-resolution deblurred images and then progressively generate high-resolution sharp results. Nah *et al.* [84] propose a multi-scale deblurring network to remove motion blur from a single image (Figure 10). In a coarse-to-fine scheme, the proposed network first generates images at 1/4 and 1/2 resolutions before estimating the deblurred image at the original scale. Numerous deep deblurring methods use a

multi-scale architecture [26, 97, 129], improving deblurring at different scales, *e.g.*, using nested skip connections in [26], or increasing the connection of networks across different scales, *e.g.*, recurrent layer applied in [129].

Reblurring networks. Reblurring networks can synthesize additional blurry training images [6, 152]. Zhang *et al.* [152] propose a framework which includes a learning-to-blur GAN (BGAN) and a learning-to-deblur GAN (DBGAN). This approach learns to transfer a sharp image to a blurry version and recover a sharp image from the blurry version, respectively. Chen *et al.* [12] introduce a reblur2deblur framework for video deblurring. Three consecutive blurry images are fed into the framework to recover sharp images, which are further used to compute the optical flow and estimate the blur kernel for reconstructing the blurry input.

Advantages and drawbacks of various models. The U-Net architecture has shown to be effective for image deblurring [112, 129] and low-level vision problems. Alternative backbone architectures for effective image deblurring include a cascade of Resblocks [57] or Dense-blocks [152]. After selecting the backbone, deep models can be improved in several ways. A multi-scale network [84] removes blur at a different scales in a coarse-to-fine way, but at increased computational cost [155]. Similarly, cascaded networks [113] recover higher-quality deblurred images via multiple deblurring stages. Deblurred images are forwarded to another network to further improve the quality. The main difference is that the deblurred images in multi-scale networks are intermediate results, whereas the output of each deblurring network in the cascaded architecture can be individually regarded as final deblurred outputs. Multi-scale architectures and cascaded networks can also be combined by treating the multi-scale networks as one stage in a cascaded network.

The primary goal for image deblurring is to achieve better contents which may be measured by full-reference metrics such as PSNR and SSIM. However, these metrics are not always consistent with human visual perception. Thus, GAN-based structures can be applied to generate realistic images [57, 58]. Within the GAN structure, all the above architectures can be employed as a generator, and a discriminator aims to make the deblurred image more realistic. For inference, only the generator is necessary. GAN-based models typically perform poorer in terms of distortion metrics such as PSNR or SSIM.

When the number of training samples is insufficient, a reblurring network may be used to generate more [152, 12]. This architecture consists of a *learn-to-blur*

and *learn-to-deblur* module. Any of the above-discussed deep models can be used to synthesize more training samples, creating training pairs of original sharp images and the output from the *learn-to-blur* model. Although this network can synthesize an unlimited number of training samples, it only models those blur effects that exist in the training samples.

The issue of pixel misalignment is challenging for image deblurring with multi-frame input. Correspondences are typically constructed by computing pixel associations in consecutive frames using optical flow or geometric transformations. For example, a pair of noisy and blurry images can be used for image deblurring and patch correspondences via optical flow [31]. When applying deep networks, alignment and deblurring can be handled jointly by providing multiple images as input and processing them via 3D convolutions [151].

5 Loss Functions

Various loss functions have been proposed to train deep deblurring networks. The pixel-wise content loss function has been widely used in the early deep deblurring networks to measure the reconstruction error. However, pixel loss cannot accurately measure the quality of deblurred images. This inspired the development of other loss functions like task-specific loss and adversarial loss for reconstructing more realistic results. In this section, we will review these loss functions.

5.1 Pixel Loss

The pixel loss function computes the pixel-wise difference between the deblurred image and the ground truth. The two main variants are the mean absolute error (L1 loss) and the mean square error (L2 loss), defined as:

$$\mathcal{L}_{pix1} = \frac{1}{WH} \sum_{x=1}^W \sum_{y=1}^H |I_{s(x,y)} - I_{db(x,y)}|, \quad (6)$$

$$\mathcal{L}_{pix2} = \frac{1}{WH} \sum_{x=1}^W \sum_{y=1}^H (I_{s(x,y)} - I_{db(x,y)})^2, \quad (7)$$

where $I_{s(x,y)}$ and $I_{db(x,y)}$ are the values of the sharp image and the deblurred image at location (x, y) , respectively. The pixel loss guides deep deblurring networks to generate sharp images close to the ground-truth pixel values. Most existing deep deblurring networks [9, 84, 129, 145, 150, 151] apply the L2 loss since it leads to a high PSNR value. Some models are trained

to optimize the L1 loss [89, 144, 152]. However, since the pixel loss function ignores long-range image structure, models trained with this loss function tend to generate over-smoothed results [58].

5.2 Perceptual Loss

Perceptual loss functions [47] have also been used to calculate the difference between images. Different from the pixel loss function, the perceptual loss compares the difference in high-level feature spaces such as the features of deep networks trained for classification, *e.g.*, VGG19 [118]. The loss is defined as:

$$\mathcal{L}_{per} = \frac{1}{WHC} \sqrt{\sum_{x=1}^W \sum_{y=1}^H \sum_{c=1}^C (\Phi_{(x,y,c)}^l(I_s) - \Phi_{(x,y,c)}^l(I_{db}))^2}, \quad (8)$$

where $\Phi_{(x,y,c)}^l(\cdot)$ are the output features of the classifier network from the l -th layer. C is the number of channels in the l -th layer. The perceptual loss function compares network features between sharp images and their deblurred versions, rather than directly matching values of each pixel, yielding visually pleasing results [57, 58, 73, 151, 152].

5.3 Adversarial Loss

For GAN-based deblurring networks, a generator network G and a discriminator network D are trained jointly such that samples generated by G can fool D . The process can be modeled as a min-max optimization problem with value function $V(G, D)$:

$$\min_G \max_D V(G, D) = E_{I \sim p_{train(I)}} [\log(D(I))] + E_{I_b \sim p_{G(I_b)}} [\log(1 - D(G(I_b)))] , \quad (9)$$

where I and I_b represent the sharp and blurry images, respectively. To guide G to generate photo-realistic sharp images, the adversarial loss function is used:

$$\mathcal{L}_{adversarial} = \log(1 - D(G(I_b))), \quad (10)$$

where $D(G(I_b))$ is the probability that the deblurred image is real. GAN-based deep deblurring methods have been applied to single image and video deblurring [57, 58, 84, 112, 151]. In contrast to the pixel and perceptual loss functions, the adversarial loss directly predicts whether the deblurred images are similar to real images and leads to photo-realistic sharp images.

5.4 Relativistic Loss

The relativistic loss is related to the adversarial loss, which can be formulated as:

$$\begin{aligned} D(I_s) &= \sigma(C(I_s)) \rightarrow 1, \\ D(I_{db}) &= D(G(I_{db})) = \sigma(C(G(I_{db}))) \rightarrow 0, \end{aligned} \quad (11)$$

where $D(\cdot)$ is the probability that the input is a real image, $C(\cdot)$ is the feature captured via a discriminator and $\sigma(\cdot)$ is the activation function. During the training stage, only the second part of Eq. 11 updates parameters of the generator G , while the first part only updates the discriminator.

To train a better generator, the relativistic loss was proposed to calculate whether a generated image is more realistic than the synthesized images [48]. Based on this loss function, Zhang *et al.* [152] replace the standard adversarial loss with the relativistic loss:

$$\begin{aligned} \sigma(C(I_s) - E(C(G(I_b)))) &\rightarrow 1, \\ \sigma(C(G(I_b)) - E(C(I_s))) &\rightarrow 0, \end{aligned} \quad (12)$$

where $E(\cdot)$ denotes the averaging operation over images in one batch. $C(\cdot)$ is the feature captured via a discriminator and $\sigma(\cdot)$ is the activation function. The generator is updated by the relativistic loss function

$$\begin{aligned} \mathcal{L}_{RDBL} &= -[\log(\sigma(C(I_r) - E(C(G(I_b)))))) \\ &\quad + \log(1 - (\sigma(G(I_b)) - E(C(I_s)))))], \end{aligned} \quad (13)$$

where I_r denotes a real image. The method provided in [58] also applies this loss function to improve the performance of image deblurring.

5.5 Optical Flow Loss

Since optical flow is able to represent the motion information between two neighboring frames, several studies remove motion blur via estimating optical flow. Gong *et al.* [29] build a CNN to first estimate the motion flow from the blurred images and then recover the deblurred images based on the estimated flow field. To obtain pairs of training samples, they simulate motion flows to generate blurred images. Chen *et al.* [12] introduce a re-blur2deblur framework, where three consecutive blurry images are input to the deblur sub-net. Then optical flow between the three deblurred images is used to estimate the blur kernel and reconstruct the input.

Advantages and drawbacks of different loss functions. Generally, all the above loss functions contribute to the progress of image deblurring. However, their characteristics and goals differ. The pixel loss generates

deblurred images which are close to the sharp ones in terms of pixel-wise measurements. Unfortunately, this typically causes over-smoothing. The perceptual loss is more consistent with human perception, while the results still exhibit obvious gaps with respect to real sharp images. The adversarial loss and optical flow loss functions aim to generate realistic deblurred images and model the motion blur, respectively. However, they cannot effectively improve the values of PSNR/SSIM or only work on motion blurred images. Multiple loss functions can also be used as a weighted sum, trading off their different properties.

6 Benchmark Datasets for Image Deblurring

In this section, we introduce public datasets for image deblurring. High-quality datasets should reflect all the different types of blur in real-world scenarios. We also introduce domain-specific datasets, *e.g.*, face and text images, applicable for domain-specific deblurring methods. Table 4 presents an overview of these datasets.

6.1 Image Deblurring Datasets

Levin *et al.* dataset. Levin *et al.* [62] develop a dataset containing 4 sharp images of size 255×255 and 8 uniform blur kernels. By convolving the real blur kernels with sharp images, this dataset contains a total of 32 synthetic blurry images.

Sun *et al.* dataset. Sun *et al.* [124] extend the dataset of Levin *et al.* [62] by using 80 high-resolution natural images from [125]. Applying the 8 blur kernels from [62], this results in 640 blurred images. Similar to Levin *et al.* [62], this dataset contains only uniformly blurred images and is insufficient for training robust CNN models.

Köhler *et al.* dataset. To simulate non-uniform blur, Köhler *et al.* [54] use a Stewart platform (*i.e.*, a robotic arm) to record the 6D camera motion and capture a printed picture. There are 4 latent sharp images and 12 camera trajectories, resulting in a total of 48 non-uniformly blurred images.

Lai *et al.* dataset. Lai *et al.* [59] provide a dataset which includes 100 real and 200 synthetic blurry images generated using both uniform blur kernels and 6D camera trajectories. The images in this dataset cover various scenarios, *e.g.*, outdoor, face, text, and low-light images, and thus can be used to evaluate deblurring methods in a variety of settings.

Table 4 Representative benchmark datasets for evaluating single image, video and domain-specific deblurring algorithms.

Dataset	Synthetic	Real	Sharp Images	Blurred Images	Blur Model	Type	Train/Test Split
Levin <i>et al.</i> [62]	✓	✗	4	32	Uniform	Single image	Not divided
Sun <i>et al.</i> [125]	✓	✗	80	640	Uniform	Single image	Not divided
Köhler <i>et al.</i> [54]	✓	✗	4	48	Non-uniform	Single image	Not divided
Lai <i>et al.</i> [59]	✓	✓	108	300	Both	Single image	Not divided
GoPro [84]	✓	✓	3,214	3,214	Non-uniform	Single image	2,103/1,111
HIDE [114]	✓	✗	8,422	8,422	Non-uniform	Single image	6,397/2,025
Blur-DVS [44]	✓	✓	2,178	2,918	Non-uniform	Single image	1,782/396
Su <i>et al.</i> [120]	✓	✓	6,708	11,925	Non-uniform	Video	5,708/1,000
REDS [83]	✓	✓	30,000	30,000	Non-uniform	Video	24,000/3,000
Hradiš <i>et al.</i> [38]	✓	✗	3M+35K	3M+35K	Non-uniform	Text	3M/35K
Shen <i>et al.</i> [112]	✓	✗	6,564	130M+16K	Uniform	Face	130M/16K
Zhou <i>et al.</i> [163]	✓	✗	20,637	20,637	Non-uniform	Stereo	17,319/3,318

GoPro dataset. Nah *et al.* [84] created a large-scale dataset to simulate real-world blur by frame averaging. A motion blurred image can be generated by integrating multiple instant and sharp images over a time interval:

$$I_b = g \left(\frac{1}{T} \int_{t=0}^T I_{s(t)} dt \right), \quad (14)$$

where $g(\cdot)$ is the Camera Response Function (CRF), and T denotes the period of camera exposure. Instead of modeling the convolution kernel, M consecutive sharp frames are averaged to generate a blurry image:

$$I_b \simeq g \left(\frac{1}{M} \sum_{t=0}^{M-1} I_{S[t]} \right). \quad (15)$$

Sharp images, captured at 240fps using a GoPro Hero4 Black camera, are averaged over time windows of varying duration to synthesize blurry images. The sharp image at the center of a time window is used as ground truth image. The GoPro dataset has been widely used to train and evaluate deep deblurring algorithms. The dataset contains 3,214 image pairs, split into training and test sets, containing 2,103 and 1,111 pairs, respectively.

HIDE dataset. Focusing mainly on pedestrians and street scenes, Shen *et al.* [114] created a motion blurred dataset, which includes camera shake and object movement. The dataset includes 6,397 and 2,025 pairs for training and testing, respectively. Similar to the GoPro dataset [84], the blurry images in the HIDE dataset are synthesized by averaging 11 continuing frames, where the central frame is used as the sharp image.

RealBlur dataset. To benchmark deep deblurring methods on real blurry images, Rim *et al.* [103] create a RealBlur dataset including two subsets. The first is RealBlur-J, which is formed with the camera JPEG

outputs. The second is RealBlur-R, which contains RAW images. The RAW images are generated by using white balance, demosaicking, and denoising operations. The dataset totally contains 9476 pairs of images.

Blur-DVS dataset. To evaluate the performance of event-based deblurring methods [44, 66], Jiang *et al.* [44] create a Blur-DVS dataset using a DAVIS240C camera. An image sequence is first captured with slow camera motion and used to synthesize 2,178 pairs of blurred and sharp images by averaging seven neighboring frames. Training and test sets contain 1,782 and 396 pairs, respectively. The dataset also provides 740 real blurry images without sharp ground-truth images.

6.2 Video Deblurring Datasets

DVD dataset. Su *et al.* [120] captured blurry video sequences with cameras on different devices, including an iPhone 5s, a Nexus 4x, and a GoPro Hero 4. To simulate the realistic motion blur, they captured sharp videos at 240 fps and then averaged eight neighboring frames to create the corresponding blurry videos at 30fps. The dataset consists of a quantitative subset and a qualitative subset. The quantitative subset includes 6,708 blurry images and their corresponding sharp images from 71 videos (61 training videos and 10 test videos). The qualitative subset covers 22 videos with a 3 – 5 second duration, which are used for visual inspection.

REDS dataset. In order to capture the motion of fast moving objects, Nah *et al.* [83] captured 300 video clips at 120 fps and 1080 × 1920 resolution using a GoPro Hero 6 Black camera, and increased the frame rate to 1920 fps by recursively applying frame interpolation [88]. By generating 24 fps blurry videos from the 1920 fps videos, spike and step artifacts present in the

DVD dataset [120] are reduced. Both the synthesized blurry frames and the sharp frames are downsampled to 720×1280 to suppress noise and compression artifacts. This dataset has also been used for evaluating single image deblurring methods [86].

6.3 Domain-Specific Datasets

Text deblurring datasets. Hradiš *et al.* [38] collected text documents from the Internet. These are downsampled to $120 - 150$ DPI resolution, and split into training and test sets of sizes 50K and 2K, respectively. Small geometric transformations with bicubic interpolation are also applied to images. Finally, 3M and 35K patches are cropped from the 50K and 2K images respectively for training and testing deblurred models. Motion blur and out-of-focus blur are used to generate the blurred images from the sharp ones. Cho *et al.* [15] also provide a text deblurring dataset with only limited number of images available.

Face deblurring datasets. Blurred face image datasets have been constructed from existing face image datasets [112, 145]. Shen *et al.* [112] collected images from the Helen [60] dataset, the CMU PIE [117] dataset, and the CelebA [69] dataset. They synthesized 20,000 motion blur kernels to generate 130 million blurry images for training, and used another 80 blur kernels to generate 16,000 images for testing.

Stereo blur dataset. Stereo cameras are widely used in fast-moving devices such as aerial vehicles and autonomous vehicles. In order to study stereo image deblurring, Zhou *et al.* [163] used a ZED stereo camera to capture 60 fps videos, increased to 480 fps via frame interpolation [88]. A varying number of successive sharp images are averaged to synthesize blurry images. This dataset includes 20,637 pairs of images, which are divided into sets of 17,319 and 3,318 for training and testing, respectively.

7 Performance Evaluation

In this section, we discuss the performance evaluation of representative deep deblurring methods.

7.1 Single Image Deblurring

We summarize representative methods in Table 5, which compares the performance on three popular single image deblurring datasets, the GoPro [84] dataset, and the datasets from Köhler *et al.* [54] and Shen *et*

al. [114]. Note that all results are obtained from the respective papers. For the results of single image deblurring on the GoPro dataset and the results of video deblurring on the DVD dataset, we additionally provide bar graphs for easier comparison, as shown in Fig. 13 and Fig. 16.

The methods developed by Sun *et al.* [123], Gong *et al.* [29] and Zhang *et al.* [150] are three early deep deblurring networks based on CNN and RNN, showing that deep learning based methods achieve competitive results. Using coarse-to-fine schemes, Nah *et al.* [84], Tao *et al.* [129], and Gao *et al.* [26] designed multi-scale networks and achieved better performance compared to single-scale deblurring networks as the coarse deblurring networks provide a better prior for higher resolutions. In addition to the multi-scale strategy, GANs have been employed to produce more realistic deblurred images [57, 58, 152]. However, GAN-based models achieve poorer results in terms of PSNR and SSIM metrics on the GoPro dataset [84] as shown in Table 5. Purohit *et al.* [96] applied an attention mechanism to the deblurring network and achieved state-of-the-art performance on the GoPro dataset. The results on Shen *et al.*'s dataset [114] demonstrate the effectiveness. Networks using ResBlocks [84] and DenseBlocks [152, 26] also achieve better performance than previous networks which directly stack convolutional layers [29, 123]. For non-UHD motion deblurring, the multi-patch architecture has the advantage of restoring images better than the multi-scale and GAN-based architectures. It can be attributed to that this architecture can pay attention to the details of each small patches. For other architectures like multi-scale networks, the small-scale version of images provides less details, and thus these methods cannot improve the performance significantly.

Fig. 12 shows sample results of single image deblurring from the GoPro dataset. We use two multi-scale methods [84, 129] and two GAN-based methods [58, 152] for experiments. Despite the differences in model architectures, all these methods perform well on this dataset. Meanwhile, methods using similar architectures can yield different results, *e.g.*, Tao *et al.* and Nah *et al.* [84], which are based on multi-scale architectures. Tao *et al.* use recurrent operation between different scales and achieve sharper deblurring results.

We also analyze the performance on the GoPro dataset using the LPIPS metric. Results of representative methods are shown in Table 6. Under this metric, Tao *et al.* [129] performs worse than Nah *et al.* [84] and Kupyn *et al.* [57]. The observation is different from Table 5 as different metrics are used. Specifically, Tao *et al.* [129] outperforms the Nah *et al.* [84] and Kupyn *et al.* [57] in terms of PSNR and SSIM on the GoPro



Fig. 12 Evaluation results of the state-of-the-art deblurring methods on the GoPro dataset [84]. From left to right: blurry images, results of Nah *et al.* [84], Tao *et al.* [129], DBGAN [152] and DeblurGAN-v2 [58]. [84] and [129] are two multi-scale based image deblurring networks. [152] and [58] are two GAN based image deblurring networks.

Table 5 Performance evaluation of representative methods for single image deblurring on three popular image deblurring datasets. MS stands for multi-scale.

Dataset	Method	Framework	Layers & block	Loss	PSNR/SSIM	Characteristic
GoPro [84]	Sun <i>et al.</i> [123]	DAE	Convolution	\mathcal{L}_{pix}	24.64/0.8429	CNN-based approach, MRF
	Gong <i>et al.</i> [29]	DAE	Fully convolution	$\mathcal{L}_{pix}, \mathcal{L}_{flow}$	26.06/0.8632	Estimation of motion flow
	Nah <i>et al.</i> [84]	MS, GAN	ResBlock	$\mathcal{L}_{per}, \mathcal{L}_{adv}$	29.23/0.9162	Multi-scale Net, adversarial loss
	Kupyn <i>et al.</i> [57]	Conditional GAN	ResBlock	$\mathcal{L}_{per}, \mathcal{L}_{adv}$	28.70/0.8580	Conditional GAN-based model
	Zhang <i>et al.</i> [150]	U-Net	Recurrent, Residual	\mathcal{L}_{pix}	29.19/0.9323	Spatially variant RNN
	Tao <i>et al.</i> [129]	U-Net, MS	Recurrent, Dense	\mathcal{L}_{pix}	30.26/0.9342	Scale-recurrent Net
	Gao <i>et al.</i> [26]	U-Net, MS	Recurrent, Dense	\mathcal{L}_{pix}	31.58/0.9478	Parameter selective sharing
	Kupyn <i>et al.</i> [58]	Conditional GAN, MS	Residual	$\mathcal{L}_{pix}, \mathcal{L}_{per}, \mathcal{L}_{adv}$	29.55/0.9340	Feature Pyramid Net, fast
	Shen <i>et al.</i> [114]	DAE	Residual, Attention	\mathcal{L}_{pix}	30.26/0.9400	Human-aware Net
	Purohit <i>et al.</i> [96]	U-Net, DAE, MS	Attention, Dense	\mathcal{L}_{pix}	32.15/0.9560	Dense deformable module
	Zhang <i>et al.</i> [152]	Reblurring, GAN	Dense	$\mathcal{L}_{pix}, \mathcal{L}_{per}, \mathcal{L}_{adv}$	31.10/0.9424	Deblurring via reblurring
	Zhang <i>et al.</i> [148]	Multi-Patch, Cascading	Residual	\mathcal{L}_{pix}	31.50/0.9483	Multi-patch Net
Köhler [54]	Jiang <i>et al.</i> [44]	DAE, Cascading	Convolution, Recurrent	$\mathcal{L}_{pix}, \mathcal{L}_{flow}, \mathcal{L}_{adv}$	31.79/0.9490	Event-based motion deblurring
	Suin <i>et al.</i> [121]	DAE	Convolution, Attention	\mathcal{L}_{pix}	32.02/0.9530	Spatially-Attentive Net
Shen [114]	Sun <i>et al.</i> [123]	DAE	Convolution	\mathcal{L}_{pix}	25.22/0.773	CNN-based approach, MRF
	Kupyn <i>et al.</i> [57]	Conditional GAN	ResBlock	$\mathcal{L}_{per}, \mathcal{L}_{adv}$	26.48/0.807	Conditional GAN-based model
	Nah <i>et al.</i> [84]	MS, GAN	ResBlock	$\mathcal{L}_{per}, \mathcal{L}_{adv}$	26.75/0.837	Multi-scale Net, adversarial loss
	Tao <i>et al.</i> [129]	U-Net, DAE, MS	Recurrent, Dense	\mathcal{L}_{pix}	26.80/0.838	Scale-recurrent Net
	Kupyn <i>et al.</i> [58]	Conditional GAN, MS	Residual	$\mathcal{L}_{pix}, \mathcal{L}_{per}, \mathcal{L}_{adv}$	26.72/0.836	Feature Pyramid Net, fast

Table 6 Performance of representative single deblurring methods using the LPIPS metric on the GoPro dataset.

Method	LPIPS
Nah <i>et al.</i> [84]	0.1819
Kupyn <i>et al.</i> [58]	0.2528
Tao <i>et al.</i> [129]	0.7879
Gao <i>et al.</i> [26]	0.0359
Zhang <i>et al.</i> [152]	0.1097

dataset. The results show that we may draw different conclusions when various metrics are used. Therefore,

it is critical to evaluate deblurring methods in different quantitative metrics. As evaluation metrics optimize different criteria, the design choice is task-dependent.

To analyze the effectiveness of different loss functions, we design a common backbone of several ResBlocks (provided by DeblurGAN-v2). Different loss functions, including L1, L2, perceptual loss, GAN loss, and RaGAN loss [152], and their various combinations are used for training. Similar to the settings in DeblurGAN-v2 [58], we train each model for 200 epochs on the GoPro dataset. The learning rate scheduler cor-

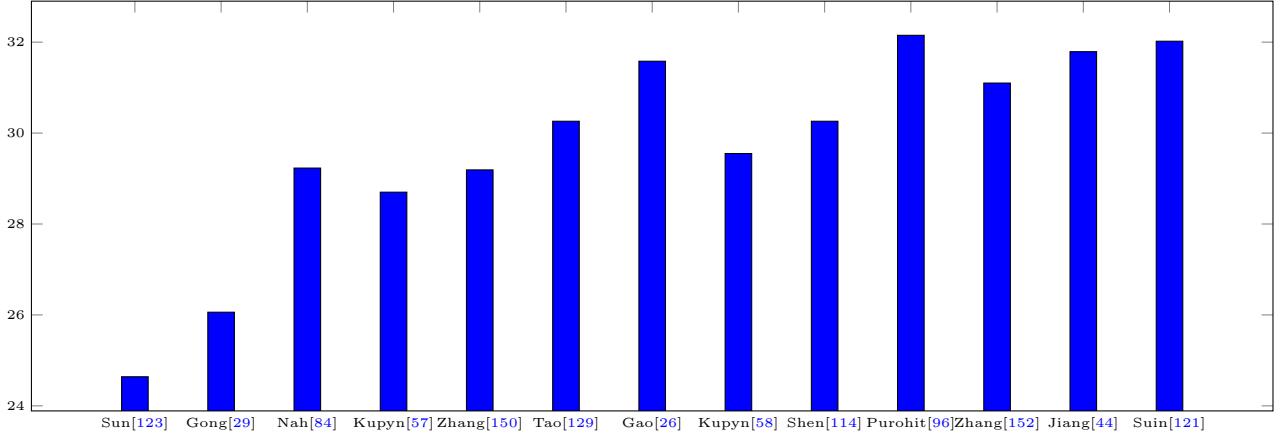


Fig. 13 Comparison among state-of-the-art image deblurring methods in terms of PSNR on the GoPro dataset.

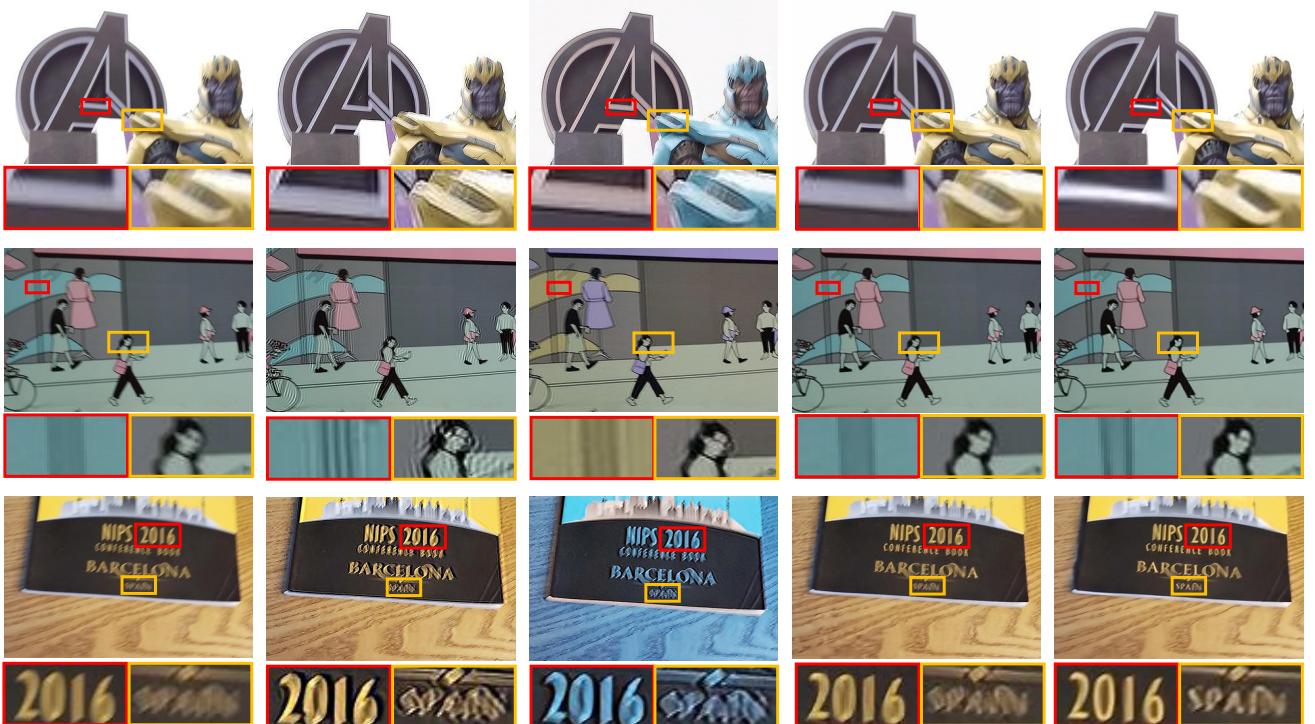


Fig. 14 Comparison among state-of-the-art non-blind and blind image deblurring methods. The blur kernels are predicted by the method in [143]

. From left to right: blurry images, results of FDN [56], RGDN [30], SRN [129] and DBGAN [152]. FDN [56] and RGDN [30] are non-blind deblurring methods, whereas SRN [129] and DBGAN [152] are blind deblurring methods.

responds to that in DeblurGAN-v2. Experimental results are reported in Table 7. In general, the evaluated models perform better when combining reconstruction and perceptual loss functions. However, using GAN-based loss functions does not necessarily improve PSNR or SSIM, which is consistent with the findings in prior reviews [61]. In addition, the same model using GAN loss or RaGAN loss achieves similar deblurring performance.

In order to evaluate the performance difference between non-blind and blind single image deblurring methods, we conduct a study comparing representative non-blind (FDN [56] and RGDN [30]) and blind single image deblurring methods [26, 58, 84, 129, 152] on the RWBI dataset. Considering that the RWBI dataset does not provide the ground-truth blur kernels, we use [143] to estimate blur kernels for non-blind image deblurring. Results are presented in Table 8 using the NIQE and BRIQUE metrics. Overall, the blind de-

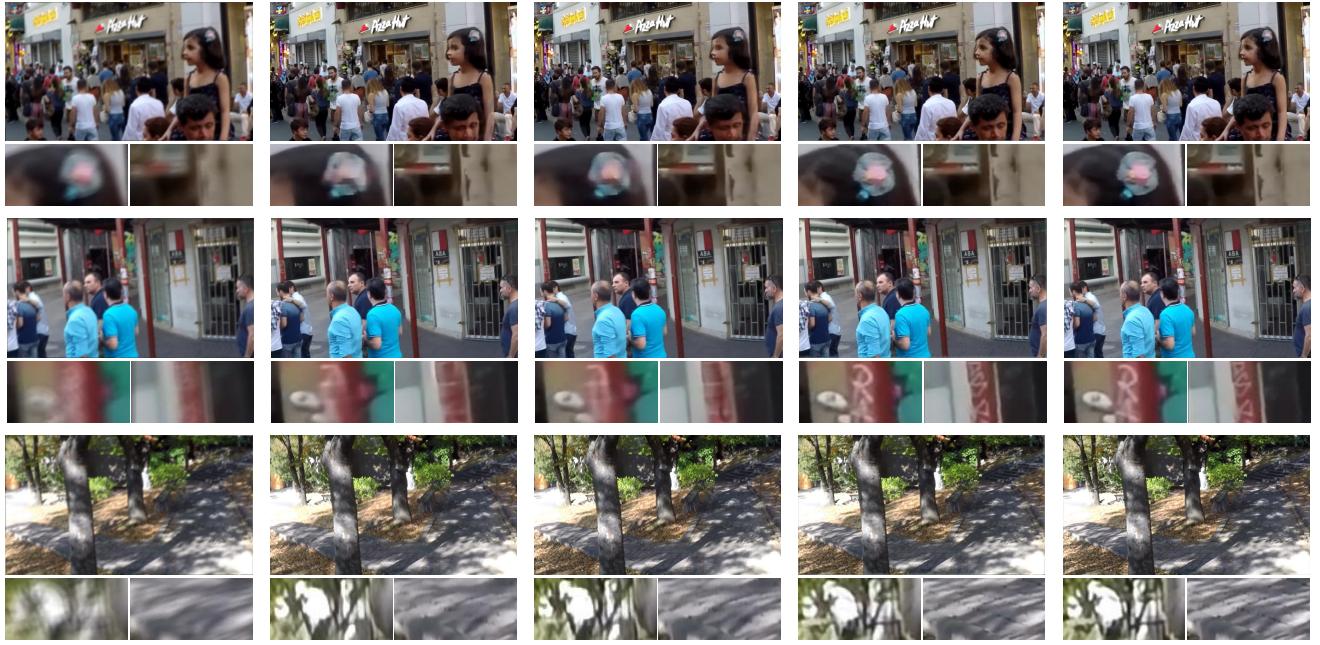


Fig. 15 Comparison among state-of-the-art non-blind and blind image deblurring methods. The ground-truth blur kernels are provided. From left to right: blurry images, results of MSCNN [84], DeblurGAN-v2 [58], DWDN [21] and USRNet [153]. MSCNN [84] and DeblurGAN-v2 [58] are blind image deblurring methods. DWDN [21] and USRNet [153] are non-blind methods.

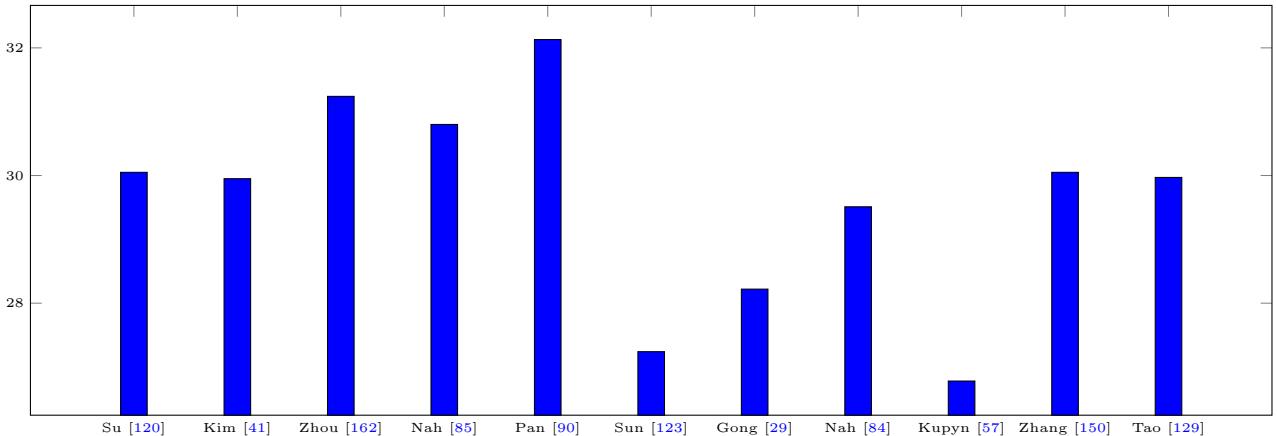


Fig. 16 Comparison among state-of-the-art video deblurring methods in terms of PSNR on the DVD dataset.

Table 7 Performance evaluation of different loss functions for single image deblurring.

Method	PSNR	SSIM
L1	26.24	0.9012
L2	26.78	0.9024
L1 + Perceptual	26.92	0.9078
L2 + Perceptual	26.95	0.9081
L1 + Perceptual + GAN	26.89	0.9060
L2 + Perceptual + GAN	27.20	0.9114
L1 + Perceptual + RaGAN	27.19	0.9111
L2 + Perceptual + RaGAN	27.09	0.9088

blurring methods outperform the non-blind methods since non-blind approaches require explicitly estimating

Table 8 The NIQE and BRISQUE of representative non-blind and blind methods for single image deblurring on the RWBI dataset. We use public available pre-trained models.

Method	NIQE	BRISQUE
FDN [56]	12.4276	57.7418
RGDN [30]	13.0038	51.5109
Nah <i>et al.</i> [84]	12.2365	49.9521
Kupyn <i>et al.</i> [58]	11.8186	40.4656
Tao <i>et al.</i> [129]	12.4606	51.1515
Gao <i>et al.</i> [26]	12.3987	50.5300
Zhang <i>et al.</i> [152]	11.5048	45.5496

the blur kernel. In practice, estimating the blur kernel is

Table 9 The PSNR and SSIM values of representative non-blind and blind deblur methods for single image deblurring on the non-blind GoPro dataset. All methods are trained on the non-blind dataset.

Method	type	PSNR	SSIM
DWDN [21]	non-blind	36.42	0.9762
USRNet [153]	non-blind	36.51	0.9823
Nah <i>et al.</i> [84]	blind	33.88	0.9795
SRN [129]	blind	34.78	0.9812

Table 10 Speed of representative methods for single image deblurring. The numbers are quoted from [121].

Method	Speed (s)
Nah <i>et al.</i> [84]	6
Kupyn <i>et al.</i> [57]	1
Tao <i>et al.</i> [129]	1.2
Zhang <i>et al.</i> [150]	1
Gao <i>et al.</i> [26]	1
Kupyn <i>et al.</i> [58]	0.48
Suin <i>et al.</i> [121]	0.34

still a challenging task. If a kernel is not well estimated, it will negatively affect the following step in recovering images. Fig. 14 shows sample results from the RWBI dataset.

To analyze the performance gap between blind and non-blind deblurring methods, we synthesize a new dataset with ground-truth blur kernels. Specifically, we use the blur kernels provided by [153] to generate blurry images based on the sharp images from the GoPro dataset [84]. The blur kernels and blurry images are the input to non-blind image deblurring methods (DWDN [21] and USRNet [153]), while the input to blind image deblurring methods (MSCNN [84] and SRN [129]) are only blurry images. Table 9 and Fig. 15 show quantitative and qualitative results, respectively. Overall, non-blind image deblurring methods can achieve better performance than blind image deblurring approaches if the ground-truth blur kernels are provided.

For a better understanding of the role of different training datasets, we train the SRN model [129] on three different public datasets (RealBlur-J [103], GoPro [84], and BSD-B [74, 103]) separately. In addition, we also train the SRN model on the combinations of “RealBlur-J + GoPro”, “RealBlur-J + BSD-B”, and “RealBlur-J + BSD-B + GoPro”. Results are reported in Table 13. The models using diverse data perform better than those only using one type of training data. In particular, the model using the combination of three datasets achieves the highest PSNR and SSIM values on the RealBlur-J testing dataset.

Considering that increasingly modern mobile devices allow capturing ultra-high-definition (UHD) images, we synthesize a new dataset to study the per-

formance of different architectures on UHD image deblurring. We use a Sony RX10 camera to capture 500 and 100 sharp images with 4K resolution as training and testing sets, respectively. Next, we use 3D camera trajectories to generate blur kernels and synthesize corresponding blurry images by convolving sharp images with large blur kernels (sizes 111×111 , 131×131 , 151×151 , 171×171 , 191×191). We use two multi-scale networks (MSCNN, SRN), two GAN-based networks (DeblurGAN, DeblurGAN-v2) and one multi-patch networks (DMPHN) for experiments. Table 14 shows the results of these representative deblurring methods. The results show that UHD image deblurring is a more challenging task and multi-scale architectures achieve better performance in terms of PSNR and SSIM. For the UHD motion deblurring, the multi-scale architectures achieve better performance. As the UHD images have higher resolution, the $1/4$ scale of them still contains fine details. The multi-scale architectures take UHD blurry images of their $1/4$ original resolution as input to generate their corresponding sharp versions. During the training stage, the images with $1/4$ resolution can provide more information for training and thus improve the performance of deblurring networks.

In order to analyze the performance of different architectures on defocus deblurring, we create another dataset and conduct numerous experiments. Specifically, we use a Sony RX10 camera to capture 500 pairs of sharp and blurry images for training, and 100 pairs of sharp and blurry images for testing. The image resolution is 900×600 pixels. Similarly, two multi-scale networks (MSCNN, SRN), two GAN-based networks (DeblurGAN, DeblurGAN-v2) and one multi-patch network (DMPHN) are evaluated on this dataset. Table 15 shows the results of the above methods. Compared with the averaging-based motion deblurring, GAN-based architectures can achieve better performance on defocus deblurring. The multi-scale and multi-patch architecture do not show significant improvement in terms of PSNR and SSIM. For the defocus deblurring, experimental results show that these two networks (with and without using the GAN framework) do not show significant difference in terms of PSNR and SSIM. However, for motion deblurring, GAN-based architectures yield lower PSNR and SSIM values. This may be attributed to that GAN-based architectures pay attention to the whole images using the adversarial loss function. In comparison, methods without GANs consider the pixel-level error (L1, L2) or perceptual loss functions, and thus ignore the whole image. We provide a run-time comparison of representative methods in Table 10.

Table 11 Performance evaluation of representative methods for video deblurring on two widely used datasets, DVD [120] and REDS [83]. MS stands for multi-scale. In order to show the performance gap between video deblurring and single image deblurring methods, the table includes some single image deblurring methods.

Dataset	Method	Framework	Layers & block	Loss	PSNR/SSIM	Characteristic
DVD [120]	Su <i>et al.</i> [120]	U-Net	Convolution	\mathcal{L}_{pix}	30.05/0.920	Video-based deep deblurring Net
	Kim <i>et al.</i> [41]	Video, DAE	Recurrent, Residual	\mathcal{L}_{pix}	29.95/0.911	Video, Dynamic temporal blending
	Zhou <i>et al.</i> [162]	DAE	Resblock	$\mathcal{L}_{pix}, \mathcal{L}_{per}$	31.24/0.934	Video, Filter Adaptive Net
	Nah <i>et al.</i> [83]	DAE	Recurrent, Resblock	$\mathcal{L}_{pix}, \mathcal{L}_{regular}$	30.80/0.8991	Video, RNN, Intra-frame iterations
	Pan <i>et al.</i> [90]	Cascade, DAE	Resblock	\mathcal{L}_{pix}	32.13/0.9268	Video, Sharpness Prior, optical flow
DVD [120]	Sun <i>et al.</i> [123]	DAE	Convolution	\mathcal{L}_{pix}	27.24/0.878	CNN-based approach, MRF
	Gong <i>et al.</i> [29]	DAE	Fully convolution	$\mathcal{L}_{pix}, \mathcal{L}_{flow}$	28.22/0.894	Estimation of motion flow
	Nah <i>et al.</i> [84]	MS, GAN	ResBlock	$\mathcal{L}_{per}, \mathcal{L}_{adv}$	29.51/0.912	Multi-scale Net, adversarial loss
	Kupyn <i>et al.</i> [57]	Conditional GAN	ResBlock	$\mathcal{L}_{per}, \mathcal{L}_{adv}$	26.78/0.848	Conditional GAN-based model
	Zhang <i>et al.</i> [150]	U-Net	Recurrent, Residual	\mathcal{L}_{pix}	30.05/0.922	Spatially variant RNN
REDS [83]	Tao <i>et al.</i> [129]	U-Net, MS	Recurrent, Dense	\mathcal{L}_{pix}	29.97/0.919	Scale-recurrent Net
	Su <i>et al.</i> [120]	U-Net	Convolution	\mathcal{L}_{pix}	26.55/0.8066	Video-based deep deblurring Net
REDS [83]	Wang <i>et al.</i> [132]	Video, DAE	Recurrent, ResBlock	\mathcal{L}_{pix}	34.80/0.9487	Video, Deformable CNN
	Kupyn <i>et al.</i> [57]	Conditional GAN	ResBlock	$\mathcal{L}_{per}, \mathcal{L}_{adv}$	24.09/0.7482	Conditional GAN-based model
	Nah <i>et al.</i> [84]	MS, GAN	ResBlock	$\mathcal{L}_{per}, \mathcal{L}_{adv}$	26.16/0.8249	Multi-scale Net, adversarial loss
	Tao <i>et al.</i> [129]	U-Net, MS	Recurrent, Dense	\mathcal{L}_{pix}	26.98/0.8141	Scale-recurrent Net

Table 12 Speed of representative methods for deep video deblurring. The numbers are quoted from [85].

Method	Speed (fps)
Su <i>et al.</i> [120]	1.72
Kim <i>et al.</i> [41]	9.24
Nah <i>et al.</i> [85]	28.8

Table 13 Performance evaluation of different training sets [74, 84, 103]. The image deblurring model is Tao *et al.* [129], and the test set is from RealBlur-J dataset [103]. The values are reported in [103].

Training set	PSNR	SSIM
RealBlur-J	31.02	0.8987
GoPro	28.56	0.8674
BSD-B	28.68	0.8675
RealBlur-J + GoPro	31.21	0.9018
RealBlur-J + BSD-B	31.30	0.9058
RealBlur-J + BSD-B + GoPro	31.37	0.9063

Table 14 Performance evaluation of representative methods for UHD image deblurring.

Method	PSNR	SSIM
DeepDeblur [84]	21.12	0.6226
DeblurGAN [57]	19.25	0.5477
SRN [129]	21.25	0.6233
DeblurGAN-v2 [58]	19.99	0.5865
DMPHN [148]	20.98	0.6217

7.2 Video Deblurring

In this section, we compare the recent video deblurring approaches on two widely used video deblurring datasets, the DVD [120] and REDS [83] datasets, see Table 11. The deep auto-encoder [120] is the most commonly used architecture for deep video deblurring methods. Similar to single image deblurring methods, the convolutional layer and ResBlocks [34] are the most important components. The recurrent structure [53] is

Table 15 Performance evaluation of representative methods for defocus deblurring.

Method	PSNR	SSIM
DeepDeblur [84]	19.78	0.7107
DeblurGAN [57]	19.12	0.6263
SRN [129]	20.45	0.7557
DeblurGAN-v2 [58]	20.02	0.6896
DMPHN [148]	20.12	0.7467

used to extract temporal information from neighboring blurry images, which is the main difference to single image deblurring networks.

Pixel-wise loss functions are employed in most video deblurring methods, but, similar to single image deblurring, perceptual loss [163] and adversarial loss [151] have also been used in video deblurring networks.

While single image deblurring methods can be applied to videos, they are outperformed by video-based methods that exploit temporal information. Su *et al.* [120] develop a blurry video dataset (DVD) and introduce a CNN-based video deblurring network, which outperforms non-deep learning based video deblurring approaches. By using intra-frame iterations, the RNN-based video deblurring network by Nah *et al.* [83] achieves better performance than Su *et al.* [120]. Zhou *et al.* [162] proposed the STFAN module, which better incorporates information from preceding frames into the deblurring process of the current frame. In addition, a filter adaptive convolutional (FAC) layer is employed for aligning the deblurred features from these frames. Pan *et al.* [90] achieve state-of-the-art performance on the DVD dataset by using a cascaded network and PWC-Net to estimate optical flow for calculating sharpness priors [122]. A DAE network uses the priors and blurry images to estimate sharp images. Run times of representative methods are shown in Table 12.



Fig. 17 Face deblurring examples. From left to right: input images, results obtained by the methods in [143], [91], [84] and [112], respectively.

8 Domain-specific Deblurring

While most deep learning based methods are designed for deblurring generic images, *i.e.* common natural and man-made scenes, some methods have studied the deblurring problem in specific domains or settings, *e.g.*, faces and texts.

8.1 Face Image Deblurring

Early face image deblurring methods used key structures of facial images [32, 91]. Recently, deep learning solutions have dominated the development of face deblurring by exploiting specific facial characteristics [45, 101, 112, 145]. Shen *et al.* [112] use semantic information to guide the process of face deblurring. Pixel-wise semantic labels are extracted via a parsing network and serve as priors to the deblurring network to restore sharp faces. Jin *et al.* [45] develop an end-to-end network with a resampling convolution operation to widen the receptive field. Chrysos *et al.* [18] propose a two-step architecture, which first restores the low frequencies and then restores the high frequencies to ensure the outputs lie on the natural image manifold. To address the task of face video deblurring, Ren *et al.* [101] explore 3D facial priors. A deep 3D face reconstruction network generates a textured 3D face for the blurry input, and a face deblurring branch recovers the sharp face under the guidance of the posed-aligned face. Figure 17 shows deblurring results from two non-deep methods [91, 143] and two deep learning based methods [84, 112].

8.2 Text Image Deblurring

Blurred text images impact the performance of optical character recognition (OCR), *e.g.*, when reading documents, displays, or street signs. Generic image deblurring methods are not well suited for text images. In early work Panci *et al.* [93] model the text image as a random field to remove blur via Bussgang algorithms [25]. Pan *et al.* [92] deblur text images via an

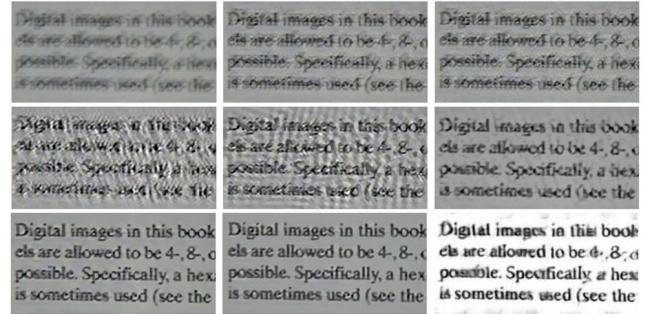


Fig. 18 Text deblurring examples. (From left to right, top to bottom) input image, results from [140] [143], [16], [160], [14], [15], [92] and [38], respectively.

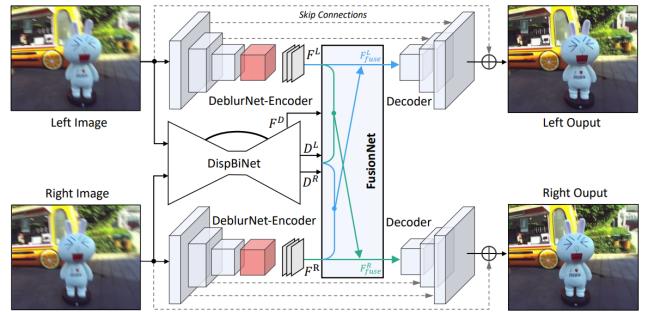


Fig. 19 Stereo image deblurring example [163]. Two stereo blurry images are fed into deep deblurring networks to generate their corresponding sharp images.

L_0 -regularized prior based on intensity and gradient. More recently, deep learning methods, *e.g.* the method by Hradiš *et al.* [38] have shown to effectively remove out-of-focus and motion blur. Xu *et al.* [145] adopt a GAN-based model to learn a category-specific prior for the task, designing a multi-class GAN model and a novel loss function for both face and text images. Figure 18 shows the deblurring results from several non-deep methods [14, 15, 16, 92, 140, 143, 160] and the deep learning based method in [38], showing that [38] achieves better deblurring results on text images.

8.3 Stereo Image Deblurring

Stereo vision has been widely used to achieve depth perception [28] and 3D scene understanding [23]. When mounting a stereo camera on a moving platform, vibration will lead to blur, negatively affecting the subsequent stereo computation [108]. To alleviate this issue, Zhou *et al.* [163] proposed a deep stereo deblurring method to make use of depth information, and varying information in corresponding pixels across two stereo images.

Depth information helps estimate the spatially-varying blur as it provides prior knowledge that nearer

points are more blurry than farther points, in the case of translational motion, which is a common real-world scenario. Further, the blur of corresponding pixels in two stereo images is different, which allows using the sharper pixel in the final deblurred images. In addition to the above tasks, other domain-specific deblurring tasks include extracting a video sequence from a single blurred image [45, 97], synthesizing high-frame-rate sharp frames [111], and joint deblurring and super-resolution [158].

9 Challenges and Opportunities

Despite the significant progress of image deblurring algorithms on benchmark datasets, recovering clear images from a real-world blurry input remains challenging [84, 120]. In this section, we summarize key limitations and discuss possible approaches and research opportunities.

Real-world data. There are three main reasons that deblurring algorithms do not perform well on real-world images.

First, most deep learning based methods require paired blurry and sharp images for training, where the blurry inputs are artificially synthesized. However, there is still a gap between these synthetic images and real-world blurry images as the blur models (*e.g.*, Eq. 3 and 15) are oversimplified [8, 12, 152]. Models trained on synthetic blur achieve excellent performance on synthetic test samples, but tend to perform worse on real-world images. One feasible approach to obtain better training samples is to build better imaging systems [103], *e.g.*, by capturing the scenes using different exposure times. Another option is to develop more realistic blur models that can synthesize more realistic training data.

Second, real-world images are not only corrupted due to blur artifacts, but also due to quantization, sensor noise, and other factors like [low-resolution](#) [112, 158]. One way to address this problem is to develop a unified image restoration model to recover high-quality images from the inputs corrupted by various nuisance factors.

Third, deblurring models trained on general images may perform poorly on images from domains that have different characteristics. Specifically, it is challenging for general methods to recover sharp images of faces or text while maintaining the identity of a particular person or characters in the text, respectively.

Loss functions. While numerous loss functions have been developed in the literature, it is not clear how to

use the right formulation for a specific scenario. For example, as Table 7 shows that an image deblurring model trained with L1 loss may achieve better performance than models trained with L2 loss on the GoPro dataset. However, the same network trained with L1 and perceptual loss functions is worse than the models trained with L2 and perceptual loss functions.

Evaluation metrics. The most widely used evaluation metrics for image deblurring are PSNR and SSIM. However, the PSNR metric is closely related to the MSE loss, which favors over-smoothed predictions. Therefore, these metrics cannot accurately reflect perceptual quality. Images with lower PSNR and SSIM values can have better visual quality [61]. The Mean Opinion Score (MOS) is an effective measure of perceptual quality. However, this metric is not universal and cannot be easily reproduced as it requires a user study. Therefore, it is still challenging to derive evaluation metrics that are consistent with the human visual response.

Data and models. Both data and models play important roles in obtaining favorable deblurring results. In the training stage, high-quality data is important to construct a strong image deblurring model. However, it is difficult to collect large-scale high-quality datasets with ground-truth. Usually, it requires using to use two cameras (with proper configurations) to capture real pairs of training samples, and thus the amount of these high-quality data is small with small diversity. While it is easier to generate a dataset with synthesized images, deblurring models developed based on such data usually perform worse than those built upon high-quality real-world samples. Therefore, constructing a large number of high-quality datasets is an important and challenging task. In addition, high-quality datasets should contain diverse scenarios, in terms of types of objects, motion, scenes, and image resolution. Deblurring network models have been mainly designed based on empirical knowledge. Recent neural architecture search methods, such as AutoML [67, 95, 165], may also be applied to the deblurring task. In addition, transformers [131] have achieved great success in various computer vision tasks. How to design more powerful backbones using Transformer may be an opportunity.

Computational cost. Since many current mobile devices support capturing 4K UHD images and videos, we test several state-of-the-art deep learning based deblurring methods. However, we found that most these deep learning based deblurring methods cannot handle 4K resolution images at high speed. For example, on a single NVIDIA Tesla V100 GPU, Tao *et al.* [129], Nah *et al.* [84] and Zhang *et al.* [152] take approximately 26.76, 28.41, 31.62 seconds, respectively, to generate a 4K

deblurred image. Therefore, efficiently restoring high-quality UHD images remains an open research topic. Note that most existing deblurring networks are evaluated on desktops or servers equipped with high-end GPUs. However, it requires significant effort to develop efficient deblurring algorithms directly on mobile devices. One recent example is the work by Kupyn *et al.* [58], which proposes a MobileNet-based model for more efficient deblurring.

Unpaired learning. Current deep deblurring methods rely on pairs of sharp images and their blurry counterparts. However, synthetically blurred images do not represent the range of real-world blur. To make use of unpaired example images, Lu *et al.* [70] and Madam *et al.* [73] recently proposed two unsupervised domain-specific deblurring models. Further improving semi-supervised or unsupervised methods to learn deblurring models appears to be a promising research direction.

References

- Abuolaim, A., Brown, M.S.: Defocus deblurring using dual-pixel data. In: European Conference on Computer Vision (2020) **1**
- Aittala, M., Durand, F.: Burst image deblurring using permutation invariant convolutional neural networks. In: European Conference on Computer Vision (2018) **6, 8**
- Aljadaany, R., Pal, D.K., Savvides, M.: Douglas-rachford networks: Learning both the image prior and data fidelity terms for blind image deconvolution. In: IEEE Conference on Computer Vision and Pattern Recognition (2019) **6, 7**
- Anwar, S., Hayder, Z., Porikli, F.: Depth estimation and blur removal from a single out-of-focus image. In: British Machine Vision Conference (2017) **3**
- Bae, S., Durand, F.: Defocus magnification. Computer Graphics Forum **26**(3), 571–579 (2007) **3**
- Bahat, Y., Efrat, N., Irani, M.: Non-uniform blind deblurring by reblurring. In: IEEE Conference on Computer Vision and Pattern Recognition (2017) **1, 10**
- Bigdeli, S.A., Zwicker, M., Favaro, P., Jin, M.: Deep mean-shift priors for image restoration. In: Advances in Neural Information Processing Systems, pp. 763–772 (2017) **4, 5**
- Brooks, T., Barron, J.T.: Learning to synthesize motion blur. In: IEEE Conference on Computer Vision and Pattern Recognition (2019) **21**
- Chakrabarti, A.: A neural approach to blind motion deblurring. In: European Conference on Computer Vision (2016) **6, 7, 11**
- Chakrabarti, A., Zickler, T., Freeman, W.T.: Analyzing spatially-varying blur. In: IEEE Conference on Computer Vision and Pattern Recognition (2010) **5**
- Chen, F., Ma, J.: An empirical identification method of gaussian blur parameter for image deblurring. IEEE Transactions on Signal Processing **57**(7), 2467–2478 (2009) **3**
- Chen, H., Gu, J., Gallo, O., Liu, M.Y., Veeraraghavan, A., Kautz, J.: Reblur2deblur: Deblurring videos via self-supervised learning. In: IEEE International Conference on Computational Photography (2018) **3, 8, 10, 12, 21**
- Chen, S.J., Shen, H.L.: Multispectral image out-of-focus deblurring using interchannel correlation. IEEE Transactions on Image Processing **24**(11), 4433–4445 (2015) **1**
- Chen, X., He, X., Yang, J., Wu, Q.: An effective document image deblurring algorithm. In: IEEE Conference on Computer Vision and Pattern Recognition (2011) **20**
- Cho, H., Wang, J., Lee, S.: Text image deblurring using text-specific properties. In: European Conference on Computer Vision (2012) **14, 20**
- Cho, S., Lee, S.: Fast motion deblurring. In: ACM SIGGRAPH Asia (2009) **1, 5, 20**
- Cho, S., Wang, J., Lee, S.: Handling outliers in non-blind image deconvolution. In: IEEE International Conference on Computer Vision (2011) **4**
- Chrysos, G.G., Favaro, P., Zafeiriou, S.: Motion deblurring of faces. International Journal of Computer Vision **127**(6-7), 801–823 (2019) **20**
- Damera-Venkata, N., Kite, T.D., Geisler, W.S., Evans, B.L., Bovik, A.C.: Image quality assessment based on a degradation model. IEEE Transactions on Image Processing **9**(4), 636–650 (2000) **4**
- Denton, E.L., Chintala, S., Fergus, R., et al.: Deep generative image models using a laplacian pyramid of adversarial networks. In: Advances in Neural Information Processing Systems (2015) **10**
- Dong, J., Roth, S., Schiele, B.: Deep wiener deconvolution: Wiener meets deep learning for image deblurring. Advances in Neural Information Processing Systems (2020) **5, 17, 18**
- Eigen, D., Puhrsch, C., Fergus, R.: Depth map prediction from a single image using a multi-scale deep network. In: Advances in Neural Information Processing Systems (2014) **10**
- Eslami, S.A., Heess, N., Weber, T., Tassa, Y., Szepesvari, D., Hinton, G.E., et al.: Attend, infer, repeat: Fast scene understanding with generative models. In: Advances in Neural Information Processing Systems (2016) **20**
- Fergus, R., Singh, B., Hertzmann, A., Roweis, S.T., Freeman, W.T.: Removing camera shake from a single photograph. In: ACM SIGGRAPH (2006) **1, 3, 5**
- Fiori, S., Uncini, A., Piazza, F.: Blind deconvolution by modified bussgang algorithm. In: The IEEE International Symposium on Circuits and Systems, vol. 3, pp. 1–4 (1999) **20**
- Gao, H., Tao, X., Shen, X., Jia, J.: Dynamic scene deblurring with parameter selective sharing and nested skip connections. In: IEEE Conference on Computer Vision and Pattern Recognition (2019) **3, 4, 5, 6, 7, 8, 10, 14, 15, 16, 17, 18**
- Gast, J., Sellent, A., Roth, S.: Parametric object motion from blur. In: IEEE Conference on Computer Vision and Pattern Recognition (2016) **5**
- Godard, C., Mac Aodha, O., Brostow, G.J.: Unsupervised monocular depth estimation with left-right consistency. In: IEEE Conference on Computer Vision and Pattern Recognition (2017) **20**
- Gong, D., Yang, J., Liu, L., Zhang, Y., Reid, I., Shen, C., Van Den Hengel, A., Shi, Q.: From motion blur to motion flow: a deep learning solution for removing heterogeneous motion blur. In: IEEE Conference on Computer Vision and Pattern Recognition (2017) **6, 12, 14, 15, 16, 17, 19**

30. Gong, D., Zhang, Z., Shi, Q., van den Hengel, A., Shen, C., Zhang, Y.: Learning deep gradient descent optimization for image deconvolution. *IEEE Transactions on Neural Networks and Learning Systems* (2020) **5**, 16, 17
31. Gu, C., Lu, X., He, Y., Zhang, C.: Blur removal via blurred-noisy image pair. *IEEE Transactions on Image Processing* **30**(11), 345–359 (2021) **11**
32. Hacohen, Y., Shechtman, E., Lischinski, D.: Deblurring by example using dense correspondence. In: *IEEE International Conference on Computer Vision* (2013) **20**
33. He, K., Gkioxari, G., Dollár, P., Girshick, R.: Mask r-cnn. In: *IEEE International Conference on Computer Vision* (2017) **2**
34. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: *IEEE Conference on Computer Vision and Pattern Recognition* (2016) **2**, 6, 19
35. Hirsch, M., Schuler, C.J., Harmeling, S., Schölkopf, B.: Fast removal of non-uniform camera shake. In: *IEEE International Conference on Computer Vision* (2011) **5**
36. Hore, A., Ziou, D.: Image quality metrics: Psnr vs. ssim. In: *IEEE International Conference on Pattern Recognition* (2010) **4**
37. Hoßfeld, T., Heegaard, P.E., Varela, M., Möller, S.: Qoe beyond the mos: an in-depth look at qoe via better metrics and their relation to mos. *Quality and User Experience* **1**(1), 2 (2016) **4**
38. Hradiš, M., Kotera, J., Žemcik, P., Šroubek, F.: Convolutional neural networks for direct text deblurring. In: *British Machine Vision Conference* (2015) **7**, 13, 14, 20
39. Hummel, R.A., Kimia, B., Zucker, S.W.: Deblurring gaussian blur. *Computer Vision, Graphics, and Image Processing* **38**(1), 66–80 (1987) **3**
40. Hyun Kim, T., Ahn, B., Mu Lee, K.: Dynamic scene deblurring. In: *IEEE International Conference on Computer Vision* (2013) **5**
41. Hyun Kim, T., Mu Lee, K., Scholkopf, B., Hirsch, M.: Online video deblurring via dynamic temporal blending network. In: *IEEE International Conference on Computer Vision* (2017) **5**, 6, 8, 9, 17, 19
42. Isola, P., Zhu, J.Y., Zhou, T., Efros, A.A.: Image-to-image translation with conditional adversarial networks. In: *IEEE Conference on Computer Vision and Pattern Recognition* (2017) **2**, 9
43. Jiang, P., Ling, H., Yu, J., Peng, J.: Salient region detection by ufo: Uniqueness, focusness and objectness. In: *IEEE International Conference on Computer Vision* (2013) **3**
44. Jiang, Z., Zhang, Y., Zou, D., Ren, J., Lv, J., Liu, Y.: Learning event-based motion deblurring. *arXiv preprint arXiv:2004.05794* (2020) **7**, 13, 15, 16
45. Jin, M., Hirsch, M., Favaro, P.: Learning face deblurring fast and wide. In: *IEEE Conference on Computer Vision and Pattern Recognition Workshop* (2018) **20**, 21
46. Jin, M., Roth, S., Favaro, P.: Noise-blind image deblurring. In: *IEEE Conference on Computer Vision and Pattern Recognition* (2017) **5**
47. Johnson, J., Alahi, A., Fei-Fei, L.: Perceptual losses for real-time style transfer and super-resolution. In: *European Conference on Computer Vision* (2016) **11**
48. Jolicoeur-Martineau, A.: The relativistic discriminator: a key element missing from standard gan. *arXiv preprint arXiv:1807.00734* (2018) **12**
49. Kang, S.B.: Automatic removal of chromatic aberration from a single image. In: *IEEE Conference on Computer Vision and Pattern Recognition* (2007) **1**
50. Kaufman, A., Fattal, R.: Deblurring using analysis-synthesis networks pair. *arXiv preprint arXiv:2004.02956* (2020) **6**, 7
51. Kettunen, M., Härkönen, E., Lehtinen, J.: E-lpis: robust perceptual image similarity via random transformation ensembles. *arXiv preprint arXiv:1906.03973* (2019) **4**
52. Kheradmand, A., Milanfar, P.: A general framework for regularized, similarity-based image restoration. *IEEE Transactions on Image Processing* **23**(12), 5136–5151 (2014) **4**
53. Kim, T.H., Sajjadi, M.S., Hirsch, M., Schölkopf, B.: Spatio-temporal transformer network for video restoration. In: *European Conference on Computer Vision* (2018) **6**, 19
54. Köhler, R., Hirsch, M., Mohler, B., Schölkopf, B., Harmeling, S.: Recording and playback of camera shake: Benchmarking blind deconvolution with a real-world database. In: *European Conference on Computer Vision* (2012) **12**, 13, 14, 15
55. Krishnan, D., Fergus, R.: Fast image deconvolution using hyper-laplacian priors. In: *Advances in Neural Information Processing Systems* (2009) **4**
56. Kruse, J., Rother, C., Schmidt, U.: Learning to push the limits of efficient fft-based image deconvolution. In: *IEEE International Conference on Computer Vision* (2017) **4**, 5, 16, 17
57. Kupyn, O., Budzan, V., Mykhailych, M., Mishkin, D., Matas, J.: Deblurgan: Blind motion deblurring using conditional adversarial networks. In: *IEEE Conference on Computer Vision and Pattern Recognition* (2018) **3**, 4, 6, 7, 9, 10, 11, 14, 15, 16, 17, 18, 19
58. Kupyn, O., Martyniuk, T., Wu, J., Wang, Z.: Deblurgan-v2: Deblurring (orders-of-magnitude) faster and better. In: *IEEE International Conference on Computer Vision* (2019) **3**, 4, 7, 9, 10, 11, 12, 14, 15, 16, 17, 18, 19, 22
59. Lai, W.S., Huang, J.B., Hu, Z., Ahuja, N., Yang, M.H.: A comparative study for single image blind deblurring. In: *IEEE Conference on Computer Vision and Pattern Recognition* (2016) **12**, 13
60. Le, V., Brandt, J., Lin, Z., Bourdev, L., Huang, T.S.: Interactive facial feature localization. In: *European Conference on Computer Vision* (2012) **14**
61. Ledig, C., Theis, L., Huszár, F., Caballero, J., Cunningham, A., Acosta, A., Aitken, A., Tejani, A., Totz, J., Wang, Z., et al.: Photo-realistic single image super-resolution using a generative adversarial network. In: *IEEE Conference on Computer Vision and Pattern Recognition* (2017) **16**, 21
62. Levin, A., Weiss, Y., Durand, F., Freeman, W.T.: Understanding and evaluating blind deconvolution algorithms. In: *IEEE Conference on Computer Vision and Pattern Recognition* (2009) **12**, 13
63. Li, L., Pan, J., Lai, W.S., Gao, C., Sang, N., Yang, M.H.: Dynamic scene deblurring by depth guided model. *IEEE Transactions on Image Processing* **29**, 5273–5288 (2020) **6**
64. Li, P., Prieto, L., Mery, D., Flynn, P.: Face recognition in low quality images: a survey. *arXiv preprint arXiv:1805.11519* (2018) **4**
65. Li, Y., Tofiqhi, M., Geng, J., Monga, V., Eldar, Y.: Deep algorithm unrolling for blind image deblurring. *arXiv preprint arXiv:1902.03493* (2019) **5**
66. Lin, S., Zhang, J., Pan, J., Jiang, Z., Zou, D., Wang, Y., Chen, J., Ren, J.: Learning event-driven video deblurring and interpolation. In: *European Conference on Computer Vision* (2020) **13**
67. Liu, H., Simonyan, K., Yang, Y.: Darts: Differentiable architecture search. In: *International Conference on Learning Representations* (2018) **21**

68. Liu, L., Liu, B., Huang, H., Bovik, A.C.: No-reference image quality assessment based on spatial and spectral entropies. *Signal Processing: Image Communication* **29**(8), 856–863 (2014) [4](#)
69. Liu, Z., Luo, P., Wang, X., Tang, X.: Deep learning face attributes in the wild. In: *IEEE International Conference on Computer Vision* (2015) [14](#)
70. Lu, B., Chen, J.C., Chellappa, R.: Unsupervised domain-specific deblurring via disentangled representations. In: *IEEE Conference on Computer Vision and Pattern Recognition* (2019) [6](#), [7](#), [22](#)
71. Lu, Y.: Out-of-focus blur: Image de-blurring. arXiv preprint arXiv:1710.00620 (2017) [3](#)
72. Lumentut, J.S., Kim, T.H., Ramamoorthi, R., Park, I.K.: Fast and full-resolution light field deblurring using a deep neural network. arXiv preprint arXiv:1904.00352 (2019) [6](#)
73. Madam Nimisha, T., Sunil, K., Rajagopalan, A.: Unsupervised class-specific deblurring. In: *European Conference on Computer Vision* (2018) [6](#), [7](#), [11](#), [22](#)
74. Martin, D., Fowlkes, C., Tal, D., Malik, J.: A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In: *IEEE International Conference on Computer Vision* (2001) [18](#), [19](#)
75. Masia, B., Corrales, A., Presa, L., Gutierrez, D.: Coded apertures for defocus deblurring. In: *Symposium Iberoamericano de Computacion Grafica* (2011) [3](#)
76. Michaeli, T., Irani, M.: Blind deblurring using internal patch recurrence. In: *European Conference on Computer Vision* (2014) [5](#)
77. Mitsa, T., Varkur, K.L.: Evaluation of contrast sensitivity functions for the formulation of quality measures incorporated in halftoning algorithms. In: *IEEE International Conference on Acoustics, Speech, and Signal Processing* (1993) [4](#)
78. Mittal, A., Moorthy, A.K., Bovik, A.C.: No-reference image quality assessment in the spatial domain. *IEEE Transactions on Image Processing* **21**(12), 4695–4708 (2012) [4](#)
79. Mittal, A., Soundararajan, R., Bovik, A.C.: Making a “completely blind” image quality analyzer. *IEEE Signal Processing Letters* **20**(3), 209–212 (2012) [4](#)
80. Moorthy, A.K., Bovik, A.C.: A two-step framework for constructing blind image quality indices. *IEEE Signal Processing Letters* **17**(5), 513–516 (2010) [4](#)
81. Moorthy, A.K., Bovik, A.C.: Blind image quality assessment: From natural scene statistics to perceptual quality. *IEEE Transactions on Image Processing* **20**(12), 3350–3364 (2011) [4](#)
82. Mustaniemi, J., Kannala, J., Särkkä, S., Matas, J., Heikkila, J.: Gyroscope-aided motion deblurring with deep networks. In: *IEEE Winter Conference on Applications of Computer Vision* (2019) [6](#), [7](#)
83. Nah, S., Baik, S., Hong, S., Moon, G., Son, S., Timofte, R., Mu Lee, K.: Ntire 2019 challenge on video deblurring and super-resolution: Dataset and study. In: *IEEE Conference on Computer Vision and Pattern Recognition Workshop* (2019) [6](#), [13](#), [19](#)
84. Nah, S., Hyun Kim, T., Mu Lee, K.: Deep multi-scale convolutional neural network for dynamic scene deblurring. In: *IEEE Conference on Computer Vision and Pattern Recognition* (2017) [3](#), [4](#), [5](#), [6](#), [7](#), [9](#), [10](#), [11](#), [13](#), [14](#), [15](#), [16](#), [17](#), [18](#), [19](#), [20](#), [21](#)
85. Nah, S., Son, S., Lee, K.M.: Recurrent neural networks with intra-frame iterations for video deblurring. In: *IEEE Conference on Computer Vision and Pattern Recognition* (2019) [6](#), [8](#), [9](#), [17](#), [19](#)
86. Nah, S., Son, S., Timofte, R., Lee, K.M.: Ntire 2020 challenge on image and video deblurring. arXiv preprint arXiv:2005.01244 (2020) [14](#)
87. Nan, Y., Quan, Y., Ji, H.: Variational-em-based deep learning for noise-blind image deblurring. In: *IEEE Conference on Computer Vision and Pattern Recognition* (2020) [4](#)
88. Niklaus, S., Mai, L., Liu, F.: Video frame interpolation via adaptive separable convolution. In: *IEEE International Conference on Computer Vision* (2017) [13](#), [14](#)
89. Nimisha, T.M., Kumar Singh, A., Rajagopalan, A.N.: Blur-invariant deep learning for blind-deblurring. In: *IEEE International Conference on Computer Vision* (2017) [6](#), [7](#), [8](#), [11](#)
90. Pan, J., Bai, H., Tang, J.: Cascaded deep video deblurring using temporal sharpness prior. In: *IEEE Conference on Computer Vision and Pattern Recognition* (2020) [8](#), [10](#), [17](#), [19](#)
91. Pan, J., Hu, Z., Su, Z., Yang, M.H.: Deblurring face images with exemplars. In: *European Conference on Computer Vision* (2014) [20](#)
92. Pan, J., Hu, Z., Su, Z., Yang, M.H.: Deblurring text images via l0-regularized intensity and gradient prior. In: *IEEE Conference on Computer Vision and Pattern Recognition* (2014) [20](#)
93. Panci, G., Campisi, P., Colonnese, S., Scarano, G.: Multichannel blind image deconvolution using the bussgang algorithm: Spatial and multiresolution approaches. *IEEE Transactions on Image Processing* **12**(11), 1324–1337 (2003) [20](#)
94. Park, P.D., Kang, D.U., Kim, J., Chun, S.Y.: Multi-temporal recurrent neural networks for progressive non-uniform single image deblurring with incremental temporal training. In: *European Conference on Computer Vision* (2020) [6](#)
95. Pham, H., Guan, M., Zoph, B., Le, Q., Dean, J.: Efficient neural architecture search via parameters sharing. In: *International Conference on Machine Learning* (2018) [21](#)
96. Purohit, K., Rajagopalan, A.: Region-adaptive dense network for efficient motion deblurring. arXiv preprint arXiv:1903.11394 (2019) [6](#), [7](#), [14](#), [15](#), [16](#)
97. Purohit, K., Shah, A., Rajagopalan, A.: Bringing alive blurred moments. In: *IEEE Conference on Computer Vision and Pattern Recognition* (2019) [10](#), [21](#)
98. Ren, D., Zhang, K., Wang, Q., Hu, Q., Zuo, W.: Neural blind deconvolution using deep priors. arXiv preprint arXiv:1908.02197 (2019) [7](#)
99. Ren, S., He, K., Girshick, R., Sun, J.: Faster r-cnn: Towards real-time object detection with region proposal networks. In: *Advances in Neural Information Processing Systems* (2015) [2](#)
100. Ren, W., Pan, J., Cao, X., Yang, M.H.: Video deblurring via semantic segmentation and pixel-wise non-linear kernel. In: *IEEE International Conference on Computer Vision* (2017) [4](#)
101. Ren, W., Yang, J., Deng, S., Wipf, D., Cao, X., Tong, X.: Face video deblurring using 3d facial priors. In: *IEEE International Conference on Computer Vision* (2019) [20](#)
102. Ren, W., Zhang, J., Ma, L., Pan, J., Cao, X., Zuo, W., Liu, W., Yang, M.H.: Deep non-blind deconvolution via generalized low-rank approximation. In: *Advances in Neural Information Processing Systems* (2018) [4](#), [5](#)

103. Rim, J., Lee, H., Won, J., Cho, S.: Real-world blur dataset for learning and benchmarking deblurring algorithms. In: European Conference on Computer Vision (2020) **5**, 13, 18, 19, 21
104. Saad, M.A., Bovik, A.C., Charrier, C.: Blind image quality assessment: A natural scene statistics approach in the dct domain. *IEEE Transactions on Image Processing* **21**(8), 3339–3352 (2012) **4**
105. Schmidt, U., Rother, C., Nowozin, S., Jancsary, J., Roth, S.: Discriminative non-blind deblurring. In: IEEE Conference on Computer Vision and Pattern Recognition (2013) **1**
106. Schuler, C.J., Christopher Burger, H., Harmeling, S., Schölkopf, B.: A machine learning approach for non-blind image deconvolution. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 1067–1074 (2013) **4**
107. Schuler, C.J., Hirsch, M., Harmeling, S., Schölkopf, B.: Learning to deblur. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **38**(7), 1439–1451 (2015) **5**, 6, 7, 9, 10
108. Sellent, A., Rother, C., Roth, S.: Stereo video deblurring. In: European Conference on Computer Vision (2016) **20**
109. Sheikh, H.R., Bovik, A.C.: Image information and visual quality. *IEEE Transactions on Image Processing* **15**(2), 430–444 (2006) **4**
110. Sheikh, H.R., Bovik, A.C., De Veciana, G.: An information fidelity criterion for image quality assessment using natural scene statistics. *IEEE Transactions on Image Processing* **14**(12), 2117–2128 (2005) **4**
111. Shen, W., Bao, W., Zhai, G., Chen, L., Min, X., Gao, Z.: Blurry video frame interpolation. In: IEEE Conference on Computer Vision and Pattern Recognition (2020) **21**
112. Shen, Z., Lai, W.S., Xu, T., Kautz, J., Yang, M.H.: Deep semantic face deblurring. In: IEEE Conference on Computer Vision and Pattern Recognition (2018) **6**, 8, 9, 10, 11, 13, 14, 20, 21
113. Shen, Z., Lai, W.S., Xu, T., Kautz, J., Yang, M.H.: Exploiting semantics for face image deblurring. *International Journal of Computer Vision* pp. 1–18 (2020) **8**, 10
114. Shen, Z., Wang, W., Lu, X., Shen, J., Ling, H., Xu, T., Shao, L.: Human-aware motion deblurring. In: IEEE International Conference on Computer Vision (2019) **4**, 6, 13, 14, 15, 16
115. Shi, J., Xu, L., Jia, J.: Discriminative blur detection features. In: IEEE Conference on Computer Vision and Pattern Recognition (2014) **3**
116. Sim, H., Kim, M.: A deep motion deblurring network based on per-pixel adaptive kernels with residual down-up and up-down modules. In: IEEE Conference on Computer Vision and Pattern Recognition Workshop (2019) **6**, 8, 9
117. Sim, T., Baker, S., Bsat, M.: The cmu pose, illumination, and expression (PIE) database. In: IEEE International Conference on Automatic Face Gesture Recognition (2002) **14**
118. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556 (2014) **2**, 11
119. Son, C.H., Park, H.M.: A pair of noisy/blurry patches-based psf estimation and channel-dependent deblurring. *IEEE Transactions on Image Processing* **57**(4), 1791–1799 (2011) **4**
120. Su, S., Delbracio, M., Wang, J., Sapiro, G., Heidrich, W., Wang, O.: Deep video deblurring for hand-held cameras. In: IEEE Conference on Computer Vision and Pattern Recognition (2017) **5**, 6, 8, 9, 13, 14, 17, 19, 21
121. Suin, M., Purohit, K., Rajagopalan, A.: Spatially-attentive patch-hierarchical network for adaptive motion deblurring. arXiv preprint arXiv:2004.05343 (2020) **4**, 7, 15, 16, 18
122. Sun, D., Yang, X., Liu, M.Y., Kautz, J.: PWC-Net: Cnns for optical flow using pyramid, warping, and cost volume. In: IEEE Conference on Computer Vision and Pattern Recognition (2018) **19**
123. Sun, J., Cao, W., Xu, Z., Ponce, J.: Learning a convolutional neural network for non-uniform motion blur removal. In: IEEE Conference on Computer Vision and Pattern Recognition (2015) **1**, 6, 7, 14, 15, 16, 17, 19
124. Sun, L., Cho, S., Wang, J., Hays, J.: Edge-based blur kernel estimation using patch priors. In: IEEE International Conference on Computational Photography (2013) **12**
125. Sun, L., Hays, J.: Super-resolution from internet-scale scene matching. In: IEEE International Conference on Computational Photography (2012) **12**, 13
126. Sun, T., Peng, Y., Heidrich, W.: Revisiting cross-channel information transfer for chromatic aberration correction. In: IEEE International Conference on Computer Vision, pp. 3248–3256 (2017) **4**
127. Szeliski, R.: Computer vision: algorithms and applications. Springer Science & Business Media (2010) **1**
128. Tang, C., Zhu, X., Liu, X., Wang, L., Zomaya, A.: Defusionnet: Defocus blur detection via recurrently fusing and refining multi-scale deep features. In: IEEE Conference on Computer Vision and Pattern Recognition (2019) **3**
129. Tao, X., Gao, H., Shen, X., Wang, J., Jia, J.: Scale-recurrent network for deep image deblurring. In: IEEE Conference on Computer Vision and Pattern Recognition (2018) **3**, 4, 5, 6, 7, 8, 10, 11, 14, 15, 16, 17, 18, 19, 21
130. Vairy, M., Venkatesh, Y.V.: Deblurring gaussian blur using a wavelet array transform. *Pattern Recognition* **28**(7), 965–976 (1995) **3**
131. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L., Polosukhin, I.: Attention is all you need. arXiv preprint arXiv:1706.03762 (2017) **21**
132. Wang, X., Chan, K.C., Yu, K., Dong, C., Change Loy, C.: EDVR: Video restoration with enhanced deformable convolutional networks. In: IEEE Conference on Computer Vision and Pattern Recognition Workshop (2019) **6**, 8, 9, 19
133. Wang, Z., Bovik, A.C.: A universal image quality index. *IEEE signal processing letters* **9**(3), 81–84 (2002) **4**
134. Wang, Z., Bovik, A.C., Sheikh, H.R., Simoncelli, E.P.: Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing* **13**(4), 600–612 (2004) **4**
135. Wang, Z., Simoncelli, E.P., Bovik, A.C.: Multiscale structural similarity for image quality assessment. In: The Asilomar Conference on Signals, Systems, and Computers (2003) **4**
136. Whyte, O., Sivic, J., Zisserman, A.: Deblurring shaken and partially saturated images. *International Journal of Computer Vision* **110**(2), 185–201 (2014) **4**
137. Whyte, O., Sivic, J., Zisserman, A., Ponce, J.: Non-uniform deblurring for shaken images. *International Journal of Computer Vision* **98**(2), 168–186 (2012) **5**

138. Wieschollek, P., Hirsch, M., Scholkopf, B., Lensch, H.: Learning blind motion deblurring. In: IEEE International Conference on Computer Vision (2017) [6](#)
139. Xia, F., Wang, P., Chen, L.C., Yuille, A.L.: Zoom better to see clearer: Human and object parsing with hierarchical auto-zoom net. In: European Conference on Computer Vision (2016) [10](#)
140. Xu, L., Jia, J.: Two-phase kernel estimation for robust motion deblurring. In: European Conference on Computer Vision (2010) [1](#), [5](#), [20](#)
141. Xu, L., Ren, J.S., Liu, C., Jia, J.: Deep convolutional neural network for image deconvolution. In: Advances in Neural Information Processing Systems (2014) [3](#), [4](#), [5](#)
142. Xu, L., Tao, X., Jia, J.: Inverse kernels for fast spatial deconvolution. In: European Conference on Computer Vision (2014) [1](#)
143. Xu, L., Zheng, S., Jia, J.: Unnatural l0 sparse representation for natural image deblurring. In: IEEE Conference on Computer Vision and Pattern Recognition (2013) [16](#), [20](#)
144. Xu, X., Pan, J., Zhang, Y.J., Yang, M.H.: Motion blur kernel estimation via deep learning. IEEE Transactions on Image Processing **27**(1), 194–205 (2017) [7](#), [11](#)
145. Xu, X., Sun, D., Pan, J., Zhang, Y., Pfister, H., Yang, M.H.: Learning to super-resolve blurry face and text images. In: IEEE International Conference on Computer Vision (2017) [11](#), [14](#), [20](#)
146. Yasarla, R., Perazzi, F., Patel, V.M.: Deblurring face images using uncertainty guided multi-stream semantic networks. arXiv preprint arXiv:1907.13106 (2019) [4](#)
147. Ye, P., Kumar, J., Kang, L., Doermann, D.: Unsupervised feature learning framework for no-reference image quality assessment. In: IEEE Conference on Computer Vision and Pattern Recognition (2012) [4](#)
148. Zhang, H., Dai, Y., Li, H., Koniusz, P.: Deep stacked hierarchical multi-patch network for image deblurring. In: IEEE Conference on Computer Vision and Pattern Recognition (2019) [6](#), [7](#), [15](#), [19](#)
149. Zhang, J., Pan, J., Lai, W.S., Lau, R.W., Yang, M.H.: Learning fully convolutional networks for iterative non-blind deconvolution. In: IEEE Conference on Computer Vision and Pattern Recognition (2017) [4](#), [5](#)
150. Zhang, J., Pan, J., Ren, J., Song, Y., Bao, L., Lau, R.W., Yang, M.H.: Dynamic scene deblurring using spatially variant recurrent neural networks. In: IEEE Conference on Computer Vision and Pattern Recognition (2018) [4](#), [7](#), [11](#), [14](#), [15](#), [16](#), [17](#), [18](#), [19](#)
151. Zhang, K., Luo, W., Zhong, Y., Ma, L., Liu, W., Li, H.: Adversarial spatio-temporal learning for video deblurring. IEEE Transactions on Image Processing **28**(1), 291–301 (2018) [6](#), [8](#), [9](#), [11](#), [19](#)
152. Zhang, K., Luo, W., Zhong, Y., Stenger, B., Ma, L., Liu, W., Li, H.: Deblurring by realistic blurring. In: IEEE Conference on Computer Vision and Pattern Recognition (2020) [3](#), [4](#), [6](#), [7](#), [10](#), [11](#), [12](#), [14](#), [15](#), [16](#), [17](#), [21](#)
153. Zhang, K., Van Gool, L., Timofte, R.: Deep unfolding network for image super-resolution. In: IEEE Conference on Computer Vision and Pattern Recognition (2020) [5](#), [17](#), [18](#)
154. Zhang, K., Zuo, W., Gu, S., Zhang, L.: Learning deep cnn denoiser prior for image restoration. In: IEEE Conference on Computer Vision and Pattern Recognition (2017) [4](#), [5](#)
155. Zhang, K., Zuo, W., Zhang, L.: Deep plug-and-play super-resolution for arbitrary blur kernels. In: IEEE Conference on Computer Vision and Pattern Recognition (2019) [6](#), [10](#)
156. Zhang, R., Isola, P., Efros, A.A., Shechtman, E., Wang, O.: The unreasonable effectiveness of deep features as a perceptual metric. In: IEEE Conference on Computer Vision and Pattern Recognition (2018) [4](#)
157. Zhang, W., Cham, W.K.: Single image focus editing. In: IEEE International Conference on Computer Vision Workshop (2009) [3](#)
158. Zhang, X., Dong, H., Hu, Z., Lai, W.S., Wang, F., Yang, M.H.: Gated fusion network for joint image deblurring and super-resolution. arXiv preprint arXiv:1807.10806 (2018) [6](#), [21](#)
159. Zhao, W., Zheng, B., Lin, Q., Lu, H.: Enhancing diversity of defocus blur detectors via cross-ensemble network. In: IEEE Conference on Computer Vision and Pattern Recognition (2019) [3](#)
160. Zhong, L., Cho, S., Metaxas, D., Paris, S., Wang, J.: Handling noise in single image deblurring using directional filters. In: IEEE Conference on Computer Vision and Pattern Recognition (2013) [20](#)
161. Zhong, Z., Gao, Y., Yinqiang, Z., Bo, Z.: Efficient spatio-temporal recurrent neural network for video deblurring. In: European Conference on Computer Vision (2020) [6](#)
162. Zhou, S., Zhang, J., Pan, J., Xie, H., Zuo, W., Ren, J.: Spatio-temporal filter adaptive network for video deblurring. In: IEEE International Conference on Computer Vision (2019) [5](#), [6](#), [8](#), [9](#), [17](#), [19](#)
163. Zhou, S., Zhang, J., Zuo, W., Xie, H., Pan, J., Ren, J.S.: Davanet: Stereo deblurring with view aggregation. In: IEEE Conference on Computer Vision and Pattern Recognition (2019) [13](#), [14](#), [19](#), [20](#)
164. Zhu, J.Y., Park, T., Isola, P., Efros, A.A.: Unpaired image-to-image translation using cycle-consistent adversarial networks. In: IEEE International Conference on Computer Vision (2017) [2](#)
165. Zoph, B., Le, Q.V.: Neural architecture search with reinforcement learning. In: International Conference on Learning Representations (2017) [21](#)
166. Zoran, D., Weiss, Y.: From learning models of natural image patches to whole image restoration. In: IEEE International Conference on Computer Vision (2011) [4](#)