



Deep Image Deblurring: A Survey

Kaihao Zhang¹ · Wenqi Ren² · Wenhan Luo² · Wei-Sheng Lai⁴ · Björn Stenger³ · Ming-Hsuan Yang⁴ · Hongdong Li¹

Received: 31 October 2020 / Accepted: 24 May 2022

© The Author(s), under exclusive licence to Springer Science+Business Media, LLC, part of Springer Nature 2022

Abstract

Image deblurring is a classic problem in low-level computer vision with the aim to recover a sharp image from a blurred input image. 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 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.

Keywords Image deblurring · Low-level vision · Image enhancement · Deep learning · Image restoration

1 Introduction

Communicated by Jiaya Jia.

✉ Ming-Hsuan Yang
mhyang@ucmerced.edu

Kaihao Zhang
kaihao.zhang@anu.edu.au

Wenqi Ren
rwq.renwenqi@gmail.com

Wenhan Luo
whluo.china@gmail.com

Wei-Sheng Lai
wlai24@ucmerced.edu

Björn Stenger
bjorn@cantab.net

Hongdong Li
hongdong.li@anu.edu.au

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 (Abuolaim and Brown 2020; Chen and Shen 2015; Kang 2007; Sun et al. 2015). Some examples are given in Fig. 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 (Schmidt et al. 2013; Szeliski 2010; Xu et al. 2014). 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, the solution is regularized using various additional constraints (Bahat et al. 2017; Cho and Lee 2009; Fergus et al. 2006; Xu and Jia 2010). While these non-deep learning

¹ Australian National University, Canberra, Australia

² Sun Yat-sen University, Guangzhou 510275, China

³ Rakuten Institute of Technology, Rakuten Group Inc., Tokyo, Japan

⁴ School of Engineering, University of California at Merced, Merced, CA, USA



Fig. 1 Examples of different blurry images. Causes for blur include **a** camera shake, **b** out-of-focus scene, **c** moving objects, and **d** multiple causes, respectively

methods show good performance in certain cases, they typically do not perform well in more complicated yet common scenarios such as strong motion blur.

Recent advances of deep learning techniques have revolutionized the field of computer vision; significant progress has been made in numerous domains, including image classification (He et al. 2016; Simonyan and Zisserman 2014) and object detection (He et al. 2017; Isola et al. 2017; Ren et al. 2015; Zhu et al. 2017). 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 as 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 Fig. 2).

- To analyze the challenges of image deblurring and discuss research opportunities.

The paper is organized as follows. In Sect. 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 Sect. 5. We introduce public benchmark datasets and evaluations in Sects. 6 and 7, respectively. In Sect. 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 Sects. 3 and 4). The goal of image deblurring is to recover a sharp image, *i.e.*, finding the inverse of the blur function, as

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

where Φ^{-1} is the deblurring model, and I_{db} denotes the deblurred image, which is the estimate of the latent sharp image I_s .

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

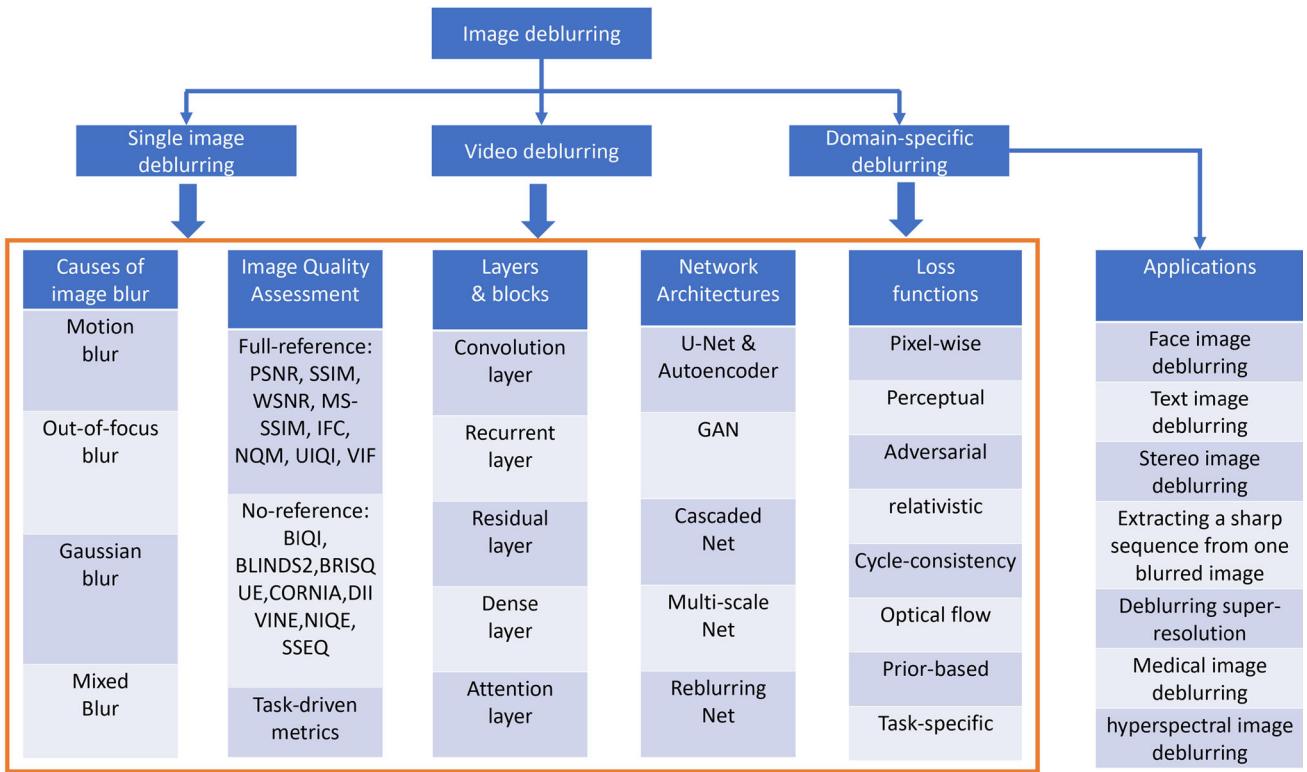


Fig. 2 Taxonomy of existing deep image deblurring techniques reviewed in this survey

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 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 (Fergus et al. 2006; Xu et al. 2014), while more recent studies address the case of non-uniform blur (Gao et al. 2019; Kupyn et al. 2018, 2019; Nah et al. 2017; Tao et al. 2018).

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. Points on the focal plane are in true focus, and points close to it appear in focus, defining the depth of field. If the scene contains objects outside this region, parts of the scene will appear blurry. The Point Spread Function (PSF) for out-of-focus blur Lu (2017) 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 Jiang et al. (2013), defocus magnification Bae

and Durand (2007) and image refocusing Zhang and Cham (2009). To address the problem of out-of-focus blur, classic methods remove blurry artifacts via blur detection Shi et al. (2014) or coded apertures Masia et al. (2011). Deep neural networks have been used to detect blur regions (Tang et al. 2019; Zhao et al. 2019) and predict depth Anwar et al. (2017) 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 methods have been developed to remove the Gaussian blur (Chen and Ma 2009; Hummel et al. 1987; Vairy and Venkatesh 1995).

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 Fig. 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 (Chen et al. 2018; Zhang et al. 2020).

In addition to these main types of blur there can be other causes, such as channel-dependent blur resulting from chromatic aberration (Son and Park 2011; Sun et al. 2017).

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) Hoßfeld et al. (2016), where people rate the quality of images on a scale of 1-5. As MOS values depend on the population sample, methods 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 Hore and Ziou (2010), SSIM Wang et al. (2004), WSNR Mitsa and Varkur (1993), MS-SSIM Wang et al. (2003), IFC Sheikh et al. (2005), NQM Damera-Venkata et al. (2000), UIQI Wang and Bovik (2002), VIF Sheikh and Bovik (2006), and LPIPS Zhang et al. (2018). Among these, PSNR and SSIM are the most commonly used metrics in image restoration tasks (Gao et al. 2019; Kupyn et al. 2018, 2019; Nah et al. 2017; Shen et al. 2019; Suin et al. 2020; Tao et al. 2018; Zhang et al. 2018a, 2020). On the other hand, LPIPS and E-LPIPS are able to approximate human judgment of image quality (Kettunen et al. 2019; Zhang et al. 2018).

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 Moorthy and Bovik (2010), BLINDS2 Saad et al. (2012), BRISQUE Mittal et al. (2012a), CORNIA Ye et al. (2012), DIIVINE Moorthy and Bovik (2011), NIQE Mittal et al. (2012b), and SSEQ Liu et al. (2014). Further, a number of metrics have been developed to evaluate the performance of image deblurring algorithms by measuring the effect on the accuracy of different vision tasks, such as object detection and recognition (Li et al. 2018; Yasarla et al. 2019).

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 Krishnan and Fergus (2009) and local Zoran and Weiss (2011) image priors, either in the spatial domain Ren et al. (2017) or in the frequency domain Kruse et al. (2017) to reconstruct sharp images. To overcome undesired ringing artifacts, Xu et al. (2014) and Ren et al. (2018) combine spatial deconvolution and deep neural networks. In addition, several approaches have been proposed to handle saturated regions (Cho et al. 2011; Whyte et al. 2014) and to remove unwanted artifacts caused by image noise (Kheradmand and Milanfar 2014; Nan et al. 2020).

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 (Ren et al. 2018; Schuler et al. 2013; Xu et al. 2014; Zhang et al. 2017, ?). Schuler et al. (2013) 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 produced in the deconvolution process. Xu et al. (2014) 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. Using the low-rank property of the pseudo-inverse kernel in Xu et al. (2014), Ren et al. (2018) 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 (Ren et al. 2018; Xu et al. 2014) 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 cannot 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 the general case. Kruse et al. (2017) propose a Fourier Deconvolution Network (FDN) by unrolling an iterative scheme, where each stage contains an FFT-based deconvolution mod-

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 Xu et al. (2014)		Gaussian, disk			First work to combine traditional optimization-based schemes and neural networks
IRCNN Zhang et al. (2017)		Gaussian, motion			Learn a set of CNN denoisers, use as a modular part of model-based optimization methods to tackle inverse problems
FCNN Zhang et al. (2017)		Motion			Adaptively learn image priors to preserve image details and structure with a robust L1 loss
FDN Kruse et al. (2017)	Uniform	Motion	Convolution	CNN	learn a CNN-based prior with an FFT-based deconvolution scheme
GLRA Ren et al. (2018)		Gaussian, disk, motion			Use generalized low-rank approximations of blur kernels to initialize the CNN parameters
DUBLID Li et al. (2019)		Motion			Recast a generalized TV-regularized algorithm into a deep network for blind image deblurring
RGDN Gong et al. (2020)		Motion			Incorporate deep neural networks into a fully parameterized gradient descent scheme
DWDN Dong et al. (2020)		Motion, Gaussian			Apply explicit deconvolution in feature space by integrating a classical Wiener deconvolution framework
USRNet Zhang et al. (2020)		Motion, Gaussian			End-to-end training of an unfolding network that integrates advantages of model-based and learning-based methods

ule and a CNN-based denoiser. Data with multiple noise levels is synthesized for training, achieving better deblurring and denoising performance.

The above methods learn denoising modules for non-blind image deblurring. Learning denoising modules can be seen as learning priors, which will be discussed in the following.

Learning Priors for Deconvolution Bigdely et al. (2017) 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 deblurring. Jin et al. (2017) use a Bayesian estimator for simultaneously estimating the noise level and removing 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. (2017) train a set of discriminative denoisers and integrate them into a model-based optimization framework to solve the non-blind deblurring problem. Without outlier handling, non-blind deblurring approaches tend to generate ringing artifacts, even in cases when the estimated kernel is accurate. Note that some super-resolution methods (with a scale factor of 1) can be adopted for the task of non-blind deblurring as the formulation as image reconstruction task is very similar, e.g., USRNet Zhang et al. (2020).

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 (Cho and Lee 2009; Fergus et al. 2006; Michaeli and Irani 2014; Schuler et al. 2015; Xu and Jia 2010). However, real-world images typically contain non-uniform blur, where different regions in the same image are generated by different blur kernels Rim et al. (2020). Numerous approaches have been developed to address non-uniform blur by modeling the blur kernel from 3D camera motion (Hirsch et al. 2011; Whyte et al. 2012). Although these approaches can model out-of-plane camera shake, they cannot handle dynamic scenes, which motivated the use of blur fields of moving objects (Chakrabarti et al. 2010; Gast et al. 2016; Hyun Kim et al. 2013). 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 (Gao et al. 2019; Nah et al. 2017; Tao et al. 2018).

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.

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 Schuler et al. (2015)		Motion		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 Hradiš et al. (2015)	Uniform	Motion & defocus	Convolution	CNN	Trains a CNN for blind deblurring and denoising.
SelfDeblur Ren et al. (2019)		Gaussian & motion		DAE	Two generative networks capture the blur kernel and a latent sharp image, respectively, which is trained on blurry images.
MRFCNN Sun et al. (2015)			Convolution	CNN	Estimate motion kernels from local patches via CNN. An MRF model predicts the motion blur field.
NDEBLUR Chakrabarti (2016)			Convolution	CNN	Train a network to generate the complex Fourier coefficients of a deconvolution filter, which is applied to the input patch.
MSCNN Nah et al. (2017)			Averaging	MS-CNN	A multi-scale CNN generates a low-resolution deblurred image and a deblurred version at the original resolution.
BIDN Nimisha et al. (2017)			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 Xu et al. (2017)			Convolution	Cascade	A two-stage CNN extracts sharp edges from blurry images for kernel estimation.
RNN_Deblur Zhang et al. (2018a)			Convolution	RNN	Deblurring via a spatially variant RNN, whose weights are learned via a CNN.
SRN Tao et al. (2018)			Averaging	MS-LSTM	Deblurring via a scale-recurrent network that shares network weights across scales.
DeblurGAN Kupyn et al. (2018)	Non-uniform	Motion	Averaging	GAN	A conditional GAN-based network generates realistic deblurred images.
UCSDBN Madam Nimisha et al. (2018)			Convolution	Cycle-GAN	An unsupervised GAN performs class-specific deblurring using unpaired images as training data.
DMPHN Zhang et al. (2019)			Convolution	DAE	A DAE network recovers sharp images based on different patches.
DeepGyro CNN Mustaniemi et al. (2019)			Convolution	DAE	A motion deblurring CNN makes use of the camera’s gyroscope readings.
PSS_SRN Gao et al. (2019)			Averaging	MS-LSTM	A selective parameter sharing scheme is applied to the SRN architecture and ResBlocks are replaced by nested skip connections.
DR_UCSDBN Lu et al. (2019)			Convolution	Cycle-GAN	Unsupervised domain-specific deblurring method by disentangling the content and blur features from input images.
Dr-Net Aljadaany et al. (2019)			Averaging	CNN	A network to learn both the image prior and data fidelity terms via Douglas-Rachford iterations.

Table 2 continued

Method	Category	Blur type	Dataset	Architecture	Key idea
DeblurGAN-v2 Kupyn et al. (2019)		Averaging	GAN		An extension of DeblurGAN using a feature pyramid network and wide range of backbone networks for better speed and accuracy.
RADN Purohit and Rajagopalan (2019)		Averaging	DAE		Region-adaptive dense deformable module to discover spatially varying shifts.
DBRBGAN Zhang et al. (2020)		Averaging	Reblur		Two networks, BGAN and DBGAN, which learn to blur and to deblur, respectively.
SAPHN Suin et al. (2020)		Averaging	DAE		Content-adaptive architecture to remove spatially-varying image blur.
ASNet Kaufman and Fattal (2020)		Convolution	DAE		DAE framework, which first estimates the blur kernel in order to recover sharp images.
EBMD Jiang et al. (2020)		Averaging	DAE		An event-based motion deblurring network, introducing a new dataset, DAVIS240C.

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. (2017), 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. (2019) and Hyun Kim et al. (2017), 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 (Chakrabarti 2016; Kaufman and Fattal 2020; Schuler et al. 2015; Sun et al. 2015) train 2D CNNs to directly recover sharp images without kernel estimation steps (Aljadaany et al. 2019; Gong et al. 2017; Lu et al. 2019; Madam Nimisha et al. 2018; Mustaniemi et al. 2019; Nimisha et al. 2017; Shen et al. 2019; Zhang et al. 2019, ?). On the other hand, several approaches use additional prior information, such as depth Li et al. (2020) or semantic labels Shen et al. (2018), to guide the deblurring process. In addition, 2D convolutions are also adopted by all video deblurring methods (Aittala and Durand 2018; Hyun Kim et al. 2017; Kim et al. 2018; Nah et al. 2019, ?; Sim

and Kim 2019; Su et al. 2017; Wang et al. 2019; Wieschollek et al. 2017). The main difference between single image and video deblurring is 3D convolutions, which can extract features from both spatial and temporal domains Zhang et al. (2018).

Recurrent Layer For single image deblurring, recurrent layers can extract features across images at multiple scales in a coarse-to-fine manner. Two representative methods are SRN Tao et al. (2018) and PSS-SRN Gao et al. (2019). SRN is 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 (Hyun Kim et al. 2017; Lumentut et al. 2019; Nah et al. 2019; Park et al. 2020; Zhong et al. 2020; Zhou et al. 2019). 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 (Hyun Kim et al. 2017; Nah et al. 2019). The second type of methods generates sharp frames via directly inputting deblurred frames from the last step Zhou et al. (2019).

Residual Layer To avoid vanishing or exploding gradients during training, the global residual layers are used to directly connect low-level and high-level layers in the area of image

Table 3 Overview of deep video deblurring methods

Method	Category	Blur type	Dataset	Architecture	Key idea
DeBlurNet Su et al. (2017)			DAE		Five neighboring blurry images are stacked and fed into a DAE to recover the center sharp image.
STRCNN Hyun Kim et al. (2017)			RNN-DAE		Recurrent architecture which includes a dynamic temporal blending mechanism for information propagation.
PICNN Aittala and Durand (2018)			DAE		Permutation invariant CNN which consists of several U-Nets taking a sequence of burst images as input.
DBLRGAN Zhang et al. (2018)		Motion	GAN		GAN-based video deblurring method, using a 3D CNN to extract spatio-temporal information.
Reblur2deblur Chen et al. (2018)	Non-uniform	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 Nah et al. (2019)			RNN-DAE		RNN-based video deblurring network, where a hidden state is transferred from past frames to the current frame.
EDVR Wang et al. (2019)			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 Zhou et al. (2019)			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 Pan et al. (2020)	Cascade				Cascaded deep video deblurring which first calculates the sharpness prior, then feeds both the blurry images and the prior into an DAE.

deblurring (Kupyn et al. 2018; Nah et al. 2017; Zhang et al. 2018). One representative method using this architecture is DeblurGAN Kupyn et al. (2018) 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 He et al. (2016), and these are widely used in image deblurring networks (Gao et al. 2019; Hyun Kim et al. 2017; Kupyn et al. 2018; Nah et al. 2017; Nimisha et al. 2017; Tao et al. 2018). Both kinds of residual layers, local and global, are often combined to achieve better performance.

Dense Layer Using dense connections can facilitate addressing the gradient vanishing problem, improving feature propagation, and reducing the number of parameters. Purohit and Rajagopalan (2019) 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. (2020) and Gao et al. (2019) 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. (2019) 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 Purohit and Rajagopalan (2019).

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. (Gao et al. 2019; Nimisha et al. 2017; Shen et al. 2018; Sim and Kim 2019; Tao et al. 2018). In some cases additional networks help exploiting additional information for guiding the U-Net. For example, Shen et al. (2018)

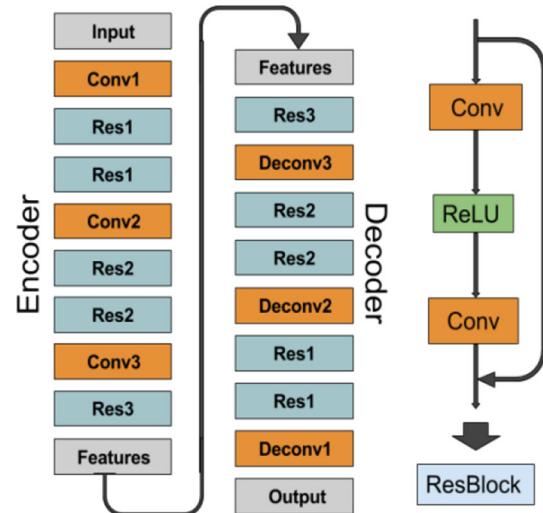


Fig. 3 Deep single image deblurring network based on the Deep Auto-Encoder (DAE) architecture Nimisha et al. (2017)

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. (2018) 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 Shen et al. (2020) combines the two ideas, using a deblurring network to obtain coarse deblurred images, then feeding 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.

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. (2017) and Wang et al. (2019) 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 Fig. 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 Zhou et al. (2019). The features from neighboring frames can also be fed into the encoder for feature extraction Hyun Kim et al. (2017). Numerous deep video deblurring algorithms based on this architecture have been developed (Nah et al. 2019; Sim and Kim 2019; Wang et al. 2019). The main differences are in the module for fusing temporal information from neigh-

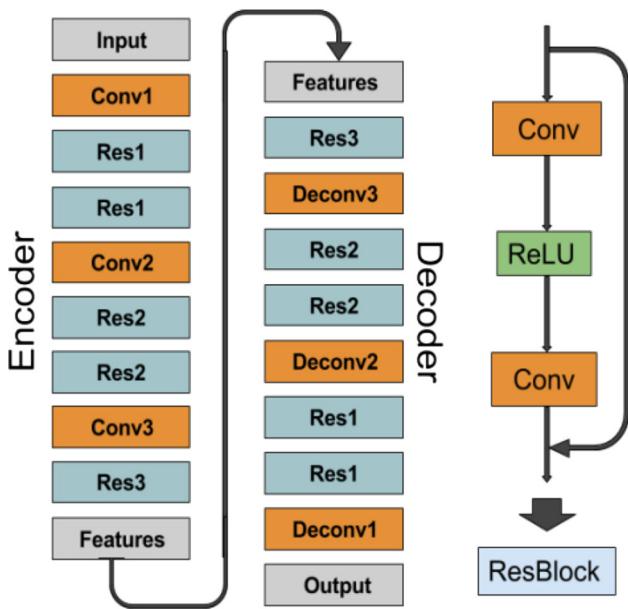


Fig. 4 Deep video deblurring network based on the DAE architecture Su et al. (2017)

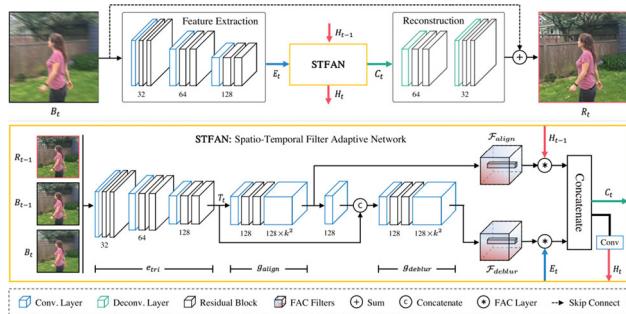


Fig. 5 Architecture to extract spatio-temporal information based on the STFAN model Zhou et al. (2019)

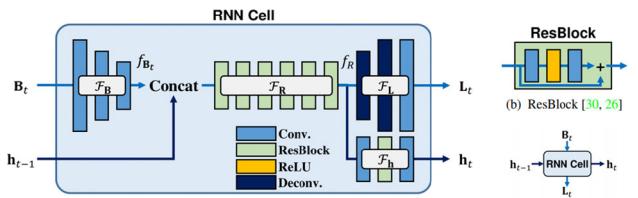


Fig. 6 Architecture to extract spatio-temporal information based on an RNN model Nah et al. (2019)

boring frames, *e.g.*, STFAN Zhou et al. (2019) in Fig. 5 and an RNN module Nah et al. (2019) in Fig. 6.

Generative Adversarial Networks (GAN) GANs have been widely used in image deblurring (Kupyn et al. 2018, 2019; Nah et al. 2017; Shen et al. 2018) in recent years. Most GAN-based deblurring models share the same strategy: the generator (Fig. 7) generates sharp images such that the discriminator cannot distinguish them from real sharp images. Kupyn et al. (2018) 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 Kupyn et al. (2019), which adopts a relativistic conditional GAN and a double-scale discriminator, which consists of local and global branches as in Isola et al. (2017). 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. (2017) and Shen et al. (2018) to generate better deblurred images.

GANs are also used in video deblurring networks. The main difference to single image deblurring is the generator, which also considers temporal information from neighboring frames. A 3DCNN model was developed by Zhang et al. (2018) to exploit both spatial and temporal information to restore sharp details using adversarial learning, see Fig. 8. Kupyn et al. (2019) proposed DeblurGAN-v2 to restore sharp

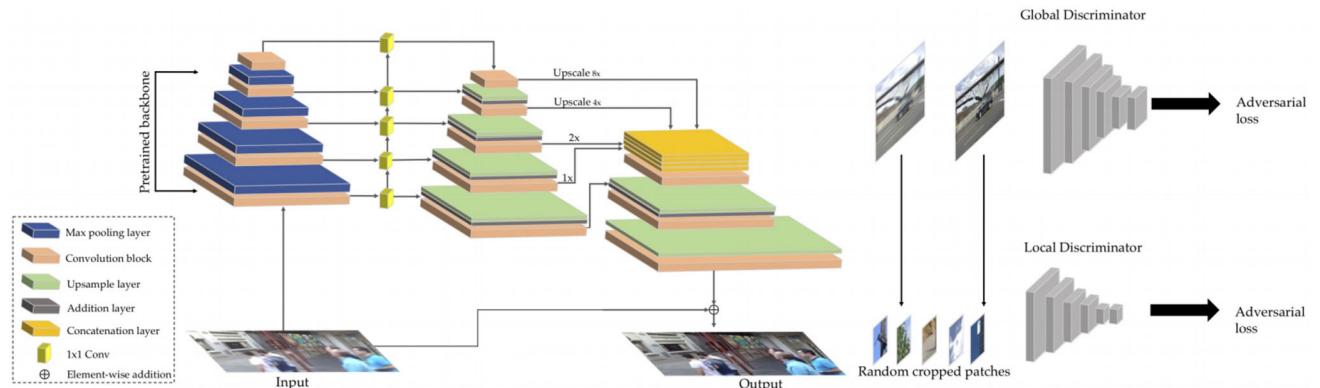


Fig. 7 Deep single image deblurring network based on the GAN architecture Kupyn et al. (2019)

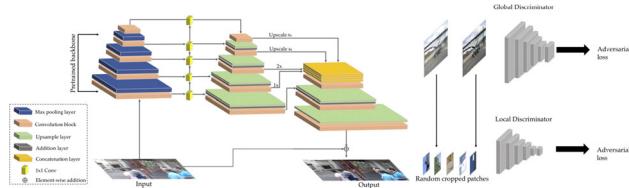


Fig. 8 Deep video deblurring network based on the GAN architecture Zhang et al. (2018)

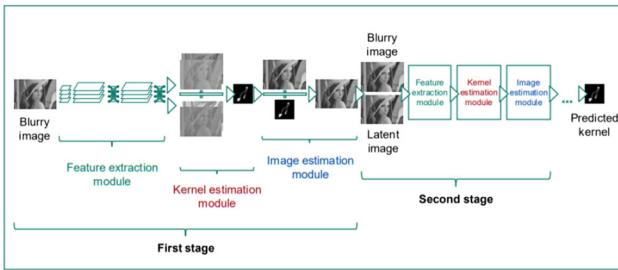


Fig. 9 Deep network for single image deblurring based on a cascaded architecture Schuler et al. (2015)

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. (2015) propose a two-stage cascaded network, see Fig. 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 deblurred 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 Pan et al. (2020).

Multi-Scale Networks Different scales of the input image describe complementary information (Denton et al. 2015; Eigen et al. 2014; Xia et al. 2016). 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. (2017) propose a multi-scale deblurring network to remove motion blur from a single image (Fig. 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 (Gao et al. 2019; Purohit et al. 2019; Tao et al. 2018),

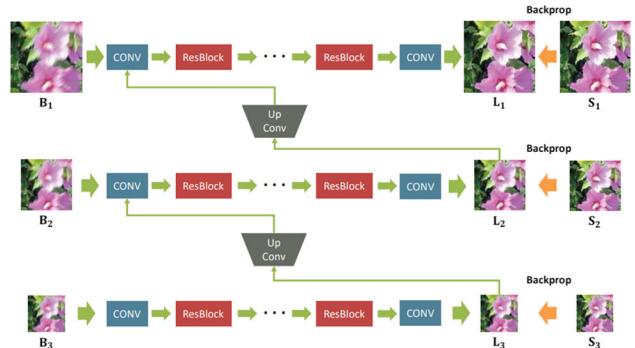


Fig. 10 Deep network for single image deblurring based on a multi-scale architecture Nah et al. (2017)

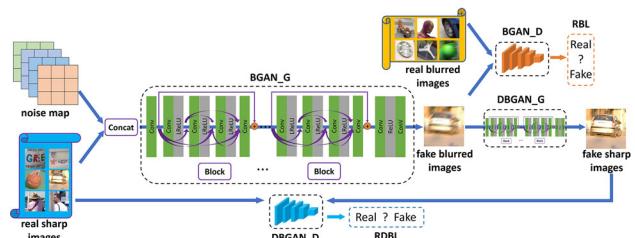


Fig. 11 Deep network for single image deblurring based on a reblurring architecture Zhang et al. (2020)

improving deblurring at different scales, *e.g.*, using nested skip connections in Gao et al. (2019), or increasing the connection of networks across different scales, *e.g.*, recurrent layers in Tao et al. (2018).

Reblurring Networks Reblurring networks can synthesize additional blurry training images (Bahat et al. 2017; Zhang et al. 2020). Zhang et al. (2020) propose a framework which includes a learning-to-blur GAN (BGAN) and a learning-to-deblur GAN (DBGAN). This approach learns to transform a sharp image into a blurry version and recover a sharp image from the blurry version, respectively. Chen et al. (2018) 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 Shen et al. (2018); Tao et al. (2018) and low-level vision problems. Alternative backbone architectures for effective image deblurring include a cascade of Resblocks Kupyn et al. (2018) or Denseblocks Zhang et al. (2020). After selecting the backbone, deep models can be improved in several ways. A multi-scale network Nah et al. (2017) removes blur at a different scales in a coarse-to-fine fashion, but at increased computational cost Zhang et al. (2019).

Similarly, cascaded networks Shen et al. (2020) 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 output. Multi-scale architectures and cascaded networks can also be combined by treating the multi-scale networks as a single stage in a cascaded network.

The primary goal of deblurring is to improve the image quality, which may be measured by reconstruction metrics such as PSNR and SSIM. However, these metrics are not always consistent with human visual perception. GANs can be trained to generate deblurred images, which are deemed realistic according to a discriminator network (Kupyn et al. 2018, 2019). 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 data (Chen et al. 2018; Zhang et al. 2020). 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 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 Gu et al. (2021). When applying deep networks, alignment and deblurring can be handled jointly by providing multiple images as input and processing them via 3D convolutions Zhang et al. (2018).

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 (Chakrabarti 2016; Nah et al. 2017; Tao et al. 2018; Xu et al. 2017; Zhang et al. 2018a, ?) apply the L2 loss since it leads to a high PSNR value. Some models are trained to optimize the L1 loss (Nimisha et al. 2017; Xu et al. 2017; Zhang et al. 2020). However, since the pixel loss function ignores long-range image structure, models trained with this loss function tend to generate over-smoothed results Kupyn et al. (2019).

5.2 Perceptual Loss

Perceptual loss functions Johnson et al (2016) 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 Simonyan and Zisserman (2014). 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 (Kupyn et al. 2018, 2019; Madam Nimisha et al. 2018; Zhang et al. 2018, 2020).

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 (Kupyn et al. 2018, 2019; Nah et al. 2017; Shen et al. 2018; Zhang et al. 2018). 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:

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

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 Jolicoeur-Martineau (2018). Based on this loss function, Zhang et al. (2020) replace the standard adversarial loss with the relativistic loss:

$$\sigma(C(I_s) - E(C(G(I_b)))) \rightarrow 1 , \\ \sigma(C(G(I_b)) - E(C(I_s)))) \rightarrow 0 , \quad (12)$$

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

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

where I_r denotes a real image. The method provided in Kupyn et al. (2019) also applies this loss function to improve the performance of image deblurring.

5.5 Optical Flow Loss

Since the 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. (2017) 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. (2018) introduce a reblur2deblur 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 To construct an image deblurring dataset, Levin et al. (2009) mount the camera on a tripod to capture blur of actual camera shake by locking the Z-axis rotation handle and allowing motion in X and Y-directions. A dataset containing 4 sharp images of size 255×255 and 8 uniform blur kernels is captured using this set-up.

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 et al. (2009)	✗	✓	4	32	Uniform	Single image	Not divided
Sun and Hays (2012)	✓	✗	80	640	Uniform	Single image	Not divided
Köhler et al. (2012)	✓	✗	4	48	Non-uniform	Single image	Not divided
Lai et al. (2016)	✓	✓	108	300	Both	Single image	Not divided
GoPro Nah et al. (2017)	✓	✓	3,214	3,214	Non-uniform	Single image	2,103/1,111
HIDE Shen et al. (2019)	✓	✗	8,422	8,422	Non-uniform	Single image	6,397/2,025
Blur-DVS Jiang et al. (2020)	✓	✓	2,178	2,918	Non-uniform	Single image	1,782/396
Su et al. (2017)	✓	✓	6,708	11,925	Non-uniform	Video	5,708/1,000
REDS Nah et al. (2019)	✓	✓	30,000	30,000	Non-uniform	Video	24,000/3,000
Hradiš et al. (2015)	✓	✗	3M+35K	3M+35K	Non-uniform	Text	3M/35K
Shen et al. (2018)	✓	✗	6,564	130M+16K	Uniform	Face	130M/16K
Zhou et al. (2019)	✓	✗	20,637	20,637	Non-uniform	Stereo	17,319/3,318

Sun et al. Dataset Sun et al. (2013) extend the dataset of Levin et al. (2009) by using 80 high-resolution natural images from Sun and Hays (2012). Applying the 8 blur kernels from Levin et al. (2009), this results in 640 blurred images. Similar to Levin et al. (2009), 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. (2012) 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. (2016) 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.

GoPro Dataset Nah et al. (2017) 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. (2019) 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 Nah et al. (2017), 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 train and benchmark deep deblurring methods on real blurry images, Rim et al. (2020) created the RealBlur dataset, consisting of two subsets. The first is RealBlur-J, which contains 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 (Jiang et al. 2020; Lin et al. 2020; Jiang et al. 2020) 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. (2017) captured blurry video sequences with cameras on different devices, including an iPhone 5s, a Nexus 4x, and a GoPro Hero 4. To simulate realistic motion blur, they captured sharp videos at 240 fps and 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. (2019) 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 Niklaus et al. (2017). By generating 24 fps blurry videos from the 1920 fps videos, spike and step artifacts present in the DVD dataset Su et al. (2017) are reduced. Both the synthesized blurry frames and the sharp frames are downscaled to 720×1280 to suppress noise and compression artifacts. This dataset has also been used for evaluating single image deblurring methods Nah et al. (2020).

6.3 Domain-Specific Datasets

Text Deblurring Datasets Hradiš et al. (2015) 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. (2012) 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 (Shen et al. 2018; Xu et al. 2017). Shen et al. (2018) collected images from the Helen Le et al. (2012) dataset, the CMU PIE Sim et al. (2002) dataset, and the CelebA Liu et al. (2015) 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. (2019) used a ZED stereo camera to capture 60 fps videos,

increased to 480 fps via frame interpolation Niklaus et al. (2017). 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 Nah et al. (2017) dataset, and the datasets from Köhler et al. (2012) and Shen et al. (2019). 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. (2015), Gong et al. (2017) and Zhang et al. (2018a) 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. (2017), Tao et al. (2018), and Gao et al. (2019) 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 (Kupyn et al. 2018, 2019; Zhang et al. 2020). However, GAN-based models achieve poorer results in terms of PSNR and SSIM metrics on the GoPro dataset Nah et al. (2017) as shown in Table 5. Purohit and Rajagopalan (2019) 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 Shen et al. (2019) demonstrate the effectiveness. Networks using ResBlocks Nah et al. (2017) and DenseBlocks (Gao et al. 2019; Zhang et al. 2020) also achieve better performance than previous networks which directly stack convolutional layers (Gong et al. 2017; Sun et al. 2015). For non-UHD motion deblurring, the multi-patch architecture has the advantage of restoring images better than the multi-scale and GAN-based architectures. For other architectures, such as multi-scale networks, the small-scale version of images provides less details, and thus these methods cannot improve the performance significantly.

Figure 12 shows sample results of single image deblurring from the GoPro dataset. We compare two multi-scale

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 Nah et al. (2017)	Sun et al. (2015)	DAE	Convolution	\mathcal{L}_{pix}	24.64/0.8429	CNN-based approach, MRF
	Gong et al. (2017)	DAE	Fully convolution	$\mathcal{L}_{pix}, \mathcal{L}_{flow}$	26.06/0.8632	Estimation of motion flow
	Nah et al. (2017)	MS, GAN	ResBlock	$\mathcal{L}_{per}, \mathcal{L}_{adv}$	29.23/0.9162	Multi-scale Net, adversarial loss
Kupyn et al. (2018)	Conditional GAN	ResBlock	$\mathcal{L}_{per}, \mathcal{L}_{adv}$	$\mathcal{L}_{per}, \mathcal{L}_{adv}$	28.7/0.8580	Conditional GAN-based model
Zhang et al. (2018a)	U-Net	Recurrent, Residual	\mathcal{L}_{pix}	\mathcal{L}_{pix}	29.19/0.9323	Spatially variant RNN
Tao et al. (2018)	U-Net, MS	Recurrent, Dense	\mathcal{L}_{pix}	\mathcal{L}_{pix}	30.26/0.9342	Scale-recurrent Net
Gao et al. (2019)	U-Net, MS	Recurrent, Dense	\mathcal{L}_{pix}	\mathcal{L}_{pix}	31.58/0.9478	Parameter selective sharing
Kupyn et al. (2019)	Conditional GAN, MS	Residual	$\mathcal{L}_{pix}, \mathcal{L}_{per}, \mathcal{L}_{adv}$	$\mathcal{L}_{pix}, \mathcal{L}_{per}, \mathcal{L}_{adv}$	29.55/0.9340	Feature Pyramid Net, fast
Shen et al. (2019)	DAE	Residual, Attention	\mathcal{L}_{pix}	\mathcal{L}_{pix}	30.26/0.9400	Human-aware Net
Purohit and Rajagopalan (2019)	U-Net, DAE, MS	Attention, Dense	\mathcal{L}_{pix}	\mathcal{L}_{pix}	32.15/0.9560	Dense deformable module
Zhang et al. (2020)	Reblurring, GAN	Dense	$\mathcal{L}_{pix}, \mathcal{L}_{per}, \mathcal{L}_{adv}$	$\mathcal{L}_{pix}, \mathcal{L}_{per}, \mathcal{L}_{adv}$	31.10/0.9424	Deblurring via reblurring
Zhang et al. (2019)	Multi-Patch, Cascading	Residual	\mathcal{L}_{pix}	\mathcal{L}_{pix}	31.5/0.9483	Multi-patch Net
Jiang et al. (2020)	DAE, Cascading	Convolution, Recurrent	$\mathcal{L}_{pix}, \mathcal{L}_{flow}, \mathcal{L}_{adv}$	$\mathcal{L}_{pix}, \mathcal{L}_{flow}, \mathcal{L}_{adv}$	31.79/0.9490	Event-based motion deblurring
Suin et al. (2020)	DAE	Convolution, Attention	\mathcal{L}_{pix}	\mathcal{L}_{pix}	32.02/0.9530	Spatially-Attentive Net
Köhler et al. (2012)	DAE	Convolution	\mathcal{L}_{pix}	\mathcal{L}_{pix}	25.22/0.773	CNN-based approach, MRF
	Kupyn et al. (2018)	Conditional GAN	ResBlock	$\mathcal{L}_{per}, \mathcal{L}_{adv}$	26.48/0.807	Conditional GAN-based model
Nah et al. (2017)	MS, GAN	ResBlock	$\mathcal{L}_{per}, \mathcal{L}_{adv}$	$\mathcal{L}_{per}, \mathcal{L}_{adv}$	26.75/0.837	Multi-scale Net, adversarial loss
Tao et al. (2018)	U-Net, DAE, MS	Recurrent, Dense	\mathcal{L}_{pix}	\mathcal{L}_{pix}	26.8/0.838	Scale-recurrent Net
Kupyn et al. (2019)	Conditional GAN, MS	Residual	$\mathcal{L}_{pix}, \mathcal{L}_{per}, \mathcal{L}_{adv}$	$\mathcal{L}_{pix}, \mathcal{L}_{per}, \mathcal{L}_{adv}$	26.72/0.836	Feature Pyramid Net, fast
Suin et al. (2015)	DAE	Convolution	\mathcal{L}_{pix}	\mathcal{L}_{pix}	23.21/0.797	CNN-based approach, MRF
Nah et al. (2017)	MS, GAN	ResBlock	$\mathcal{L}_{per}, \mathcal{L}_{adv}$	$\mathcal{L}_{per}, \mathcal{L}_{adv}$	27.43/0.902	Multi-scale Net, adversarial loss
Tao et al. (2018)	U-Net, DAE, MS	Recurrent, Dense	\mathcal{L}_{pix}	\mathcal{L}_{pix}	28.6/0.941	Scale-recurrent Net
Shen et al. (2019)	DAE	Residual, Attention	\mathcal{L}_{pix}	\mathcal{L}_{pix}	29.6/0.941	Human-aware Net



Fig. 12 Evaluation results of the state-of-the-art deblurring methods on the GoPro dataset Nah et al. (2017). From left to right: blurry images, results of Nah et al. (2017), Tao et al. (2018), DBGAN Zhang et al. (2020) and DeblurGAN-v2 Kupyn et al. (2019). Nah et al. (2017) and

Tao et al. (2018) are two multi-scale based image deblurring networks. Zhang et al. (2020) and Kupyn et al. (2019) are two GAN based image deblurring networks

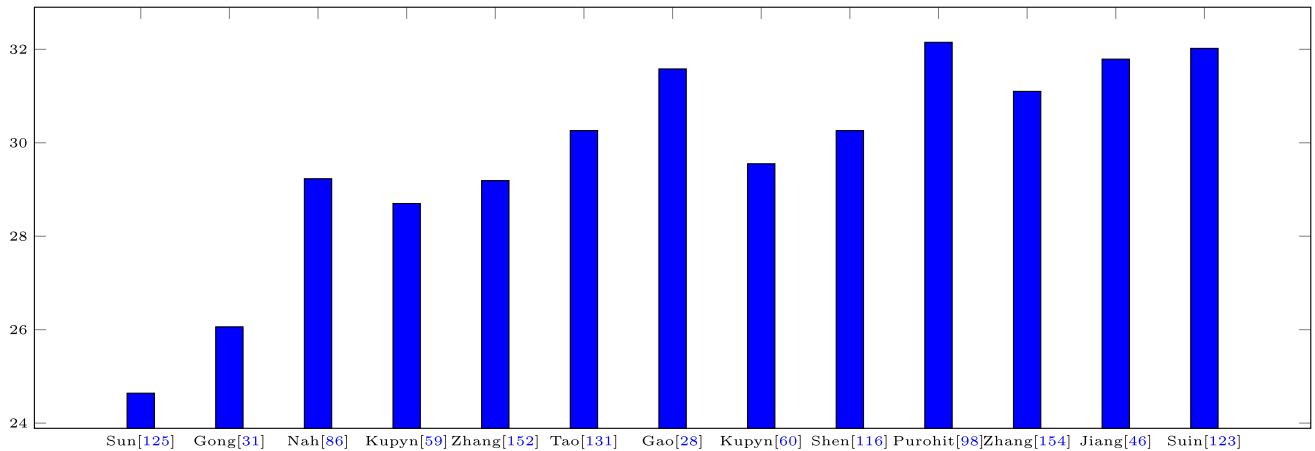


Fig. 13 Comparison among state-of-the-art image deblurring methods in terms of PSNR on the GoPro dataset

methods (Nah et al. 2017; Tao et al. 2018) and two GAN-based methods (Kupyn et al. 2019; Zhang et al. 2020) in the 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. (2017), which are based on multi-scale architectures. Tao et al. use recurrent operations 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. (2018) performs worse than Nah et al. (2017) and Kupyn et al. (2018). This is because the LPIPS measures the perceptual similarity rather than the pixel-wise similarity measures like L1. Specifically, Tao et al. (2018) outperforms the Nah et al. (2017) and Kupyn et al. (2018) in terms of PSNR and SSIM on the GoPro dataset. The results show that we may draw different conclusions when various metrics are used.

Therefore, it is critical to evaluate deblurring methods using different quantitative metrics. As evaluation metrics optimize different criteria, the design choice is task-dependent. There is a perception *vs.* distortion tradeoff in image reconstruction tasks, and it has been shown that these two measures are at odds with each other, see Blau and Michaeli (2018). To evaluate methods fairly, one may design a combined cost function including measures such as PSNR, SSIM, FID, LPIPS, NIQE, and carry out user studies.

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 Zhang et al. (2020), and their various combinations are used for training. Similar to the settings in DeblurGAN-v2 Kupyn et al. (2019), we train each model for 200 epochs on the GoPro dataset. The learning rate scheduler corresponds 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 Ledig et al. (2017). 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 Kruse et al. 2017 and RGDN Gong et al. 2020) and blind single image deblurring methods (Gao et al. 2019; Kupyn et al. 2019; Nah et al. 2017; Tao et al. 2018; Zhang et al. 2020) on the RWBI dataset. Considering that the RWBI dataset does not provide the ground-truth blur kernels, we use Xu et al. (2013) to estimate blur kernels for non-blind image deblurring. Results are presented in Table 8 using the NIQUE and BRIQUE metrics. Overall, the blind deblurring methods outperform the non-blind methods since non-blind approaches require explicitly estimating the blur kernel. In practice, estimating the blur kernel is still a challenging task. If a kernel is not well estimated, it will negatively affect the image restoration task. Figure 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 applied in Zhang et al. (2020) (including 4 isotropic Gaussian kernels, 4 anisotropic Gaussian kernels from Zhang et al. (2018), and 4 motion blur kernels from Boracchi and Foi 2012; Levin et al. 2009) to generate blurry images based on the sharp images from the GoPro dataset Nah et al. (2017). The generation method is based on Eq. 3 using Gaussian noise. The blur kernels and blurry images are the input to non-blind image deblurring methods (DWDN Dong et al. 2020)

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

Method	LPIPS
Nah et al. (2017)	0.1819
Kupyn et al. (2019)	0.2528
Tao et al. (2018)	0.7879
Gao et al. (2019)	0.0359
Zhang et al. (2020)	0.1097

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

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	NIQUE	BRISQUE
FDN Kruse et al. (2017)	12.4276	57.7418
RGDN Gong et al. (2020)	13.0038	51.5109
Nah et al. (2017)	12.2365	49.9521
Kupyn et al. (2019)	11.8186	40.4656
Tao et al. (2018)	12.4606	51.1515
Gao et al. (2019)	12.3987	50.5300
Zhang et al. (2020)	11.5048	45.5496

and USRNet Zhang et al. (2020)), while the input to blind image deblurring methods (MSCNN Nah et al. (2017) and SRN Tao et al. (2018)) 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 Tao et al. (2018) on three different public datasets (RealBlur-J Rim et al. 2020, GoPro Nah et al. 2017, and BSD-B Martin et al. 2001; Rim et al. 2020) 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

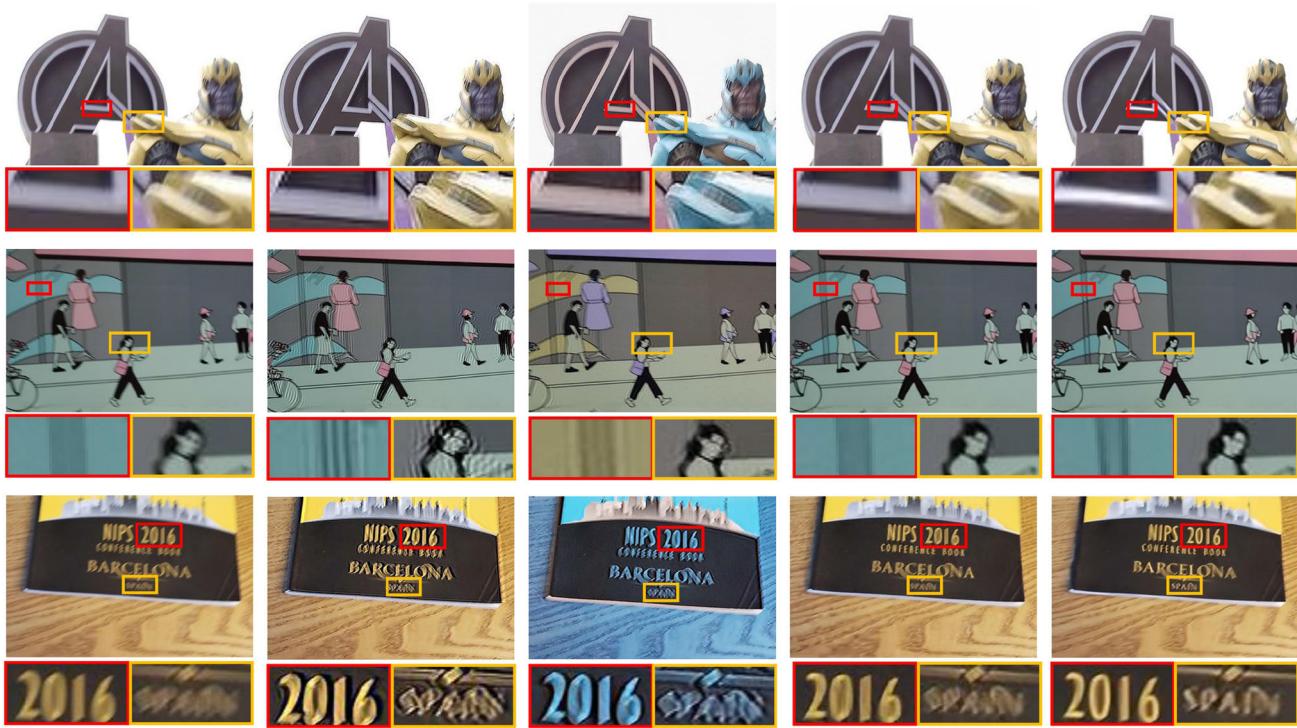


Fig. 14 Comparison among state-of-the-art non-blind and blind image deblurring methods. The blur kernels are predicted by the method in Xu et al. (2013). From left to right: blurry images, results of FDN Kruse et al. (2017), RGDN Gong et al. (2020), SRN Tao et al. (2018) and

DBGAN Zhang et al. (2020). FDN Kruse et al. (2017) and RGDN Gong et al. (2020) are non-blind deblurring methods, whereas SRN Tao et al. (2018) and DBGAN Zhang et al. (2020) are blind deblurring methods

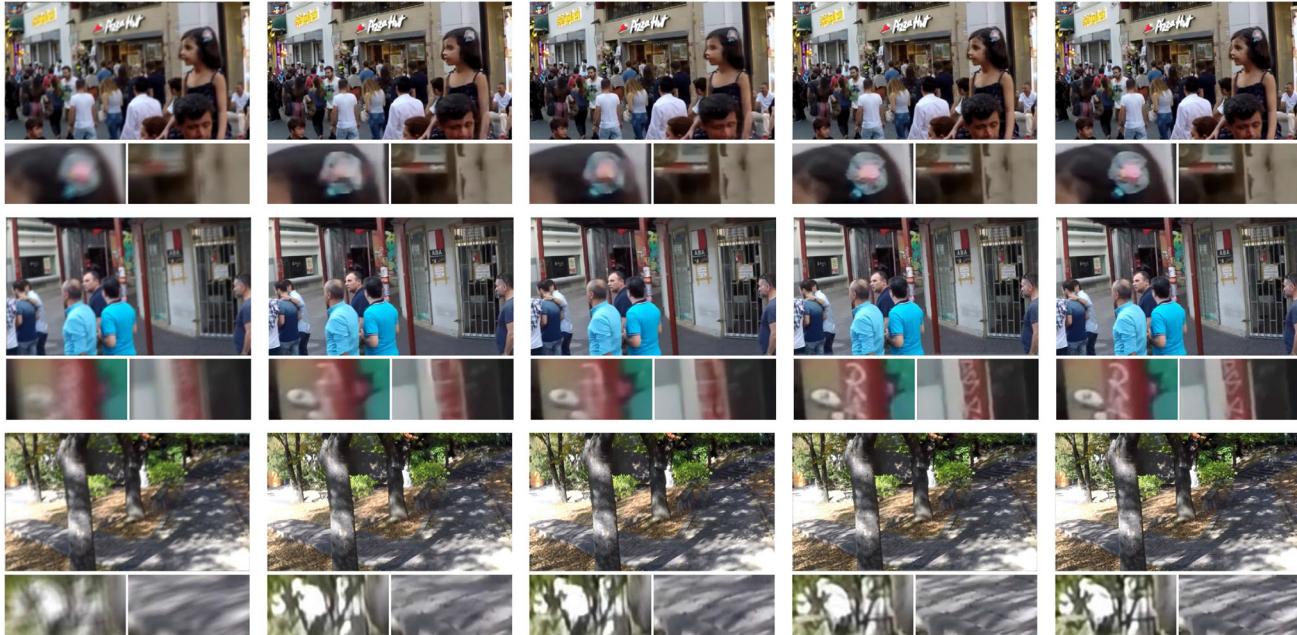


Fig. 15 Comparison among state-of-the-art non-blind and blind image deblurring methods. The ground-truth blur kernels are provided. From left to right: blurry images, results of MSCNN Nah et al. (2017), DeblurGAN-v2 Kupyn et al. (2019), DWDN Dong et al. (2020)

and USRNet Zhang et al. (2020). MSCNN Nah et al. (2017) and DeblurGAN-v2 Kupyn et al. (2019) are blind image deblurring methods. DWDN Dong et al. (2020) and USRNet Zhang et al. (2020) are non-blind methods

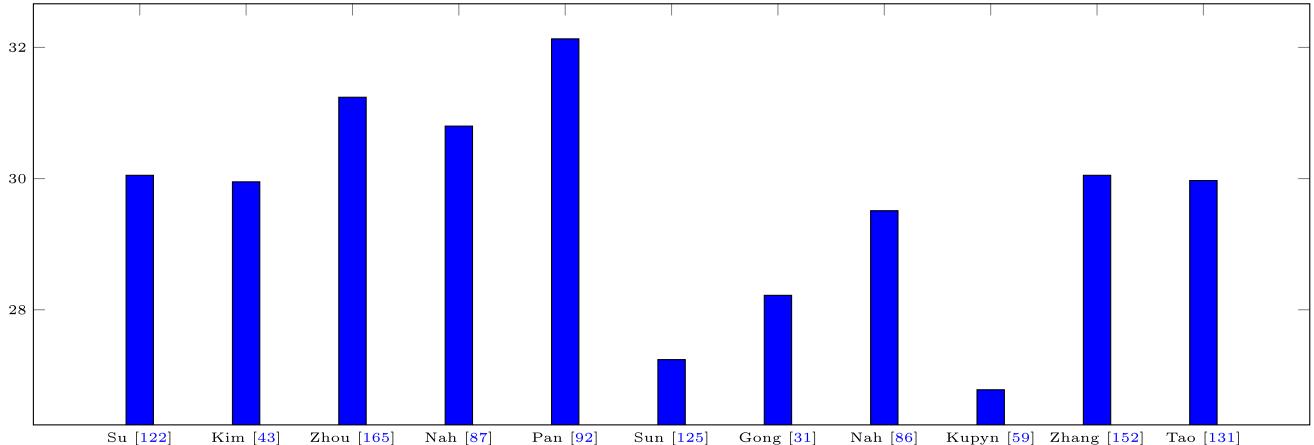


Fig. 16 Comparison among state-of-the-art video deblurring methods in terms of PSNR on the DVD dataset

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 Dong et al. (2020)	Non-blind	36.42	0.9762
USRNet Zhang et al. (2020)	Non-blind	36.51	0.9823
Nah et al. (2017)	blind	33.88	0.9795
SRN Tao et al. (2018)	blind	34.78	0.9812

Table 10 Speed of representative methods for single image deblurring. The numbers are quoted from Suin et al. (2020)

Method	Speed (s)
Nah et al. (2017)	6
Kupyn et al. (2018)	1
Tao et al. (2018)	1.2
Zhang et al. (2018a)	1
Gao et al. (2019)	1
Kupyn et al. (2019)	0.48
Suin et al. (2020)	0.34

achieves the highest PSNR and SSIM values on the RealBlur-J testing dataset.

Considering that modern mobile devices allow capturing ultra-high-definition (UHD) images, we synthesize a new dataset to study the performance of different architectures on UHD image deblurring. 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 averaging-based motion deblurring, GAN-based architectures can achieve better performance on defocus deblurring. Multi-scale and multi-patch architectures do not show significant improvements in terms of PSNR and SSIM. For defocus deblurring, results show that these two networks (with and without GAN framework) do not show significant differences in terms of PSNR and SSIM. However, for motion deblurring, GAN-based architectures yield lower PSNR and SSIM values. This may be attributed to the fact that GAN-based architectures paying attention to whole images using the adversarial loss function. In comparison, methods without GANs consider the pixel-level error (L1, L2), and thus

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 UHD motion deblurring, multi-scale architectures achieve better performance. As UHD images have higher resolution, downsampled versions at 1/4 scale still contain sufficient detail. Multi-scale architectures take UHD blurry images at 1/4 of the original resolution as input and generate the corresponding sharp versions. During the training stage, the images with 1/4 resolution can provide additional 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 averaging-based motion deblurring, GAN-based architectures can achieve better performance on defocus deblurring. Multi-scale and multi-patch architectures do not show significant improvements in terms of PSNR and SSIM. For defocus deblurring, results show that these two networks (with and without GAN framework) do not show significant differences in terms of PSNR and SSIM. However, for motion deblurring, GAN-based architectures yield lower PSNR and SSIM values. This may be attributed to the fact that GAN-based architectures paying attention to whole images using the adversarial loss function. In comparison, methods without GANs consider the pixel-level error (L1, L2), and thus

Table 11 Performance evaluation of representative methods for video deblurring on two widely used datasets, DVD Su et al. (2017) and REDS Nah et al. (2019). 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 Su et al. (2017)	Su et al. (2017)	U-Net	Convolution	\mathcal{L}_{pix}	30.05/0.920	Video-based deep deblurring Net
	Hyun Kim et al. (2017)	Video, DAE	Recurrent, Residual	\mathcal{L}_{pix}	29.95/0.911	Video, Dynamic temporal blending
	Zhou et al. (2019)	DAE	Resblock	$\mathcal{L}_{pix}, \mathcal{L}_{per}$	31.24/0.934	Video, Filter Adaptive Net
	Nah et al. (2019)	DAE	Recurrent, Resblock	$\mathcal{L}_{pix}, \mathcal{L}_{regular}$	30.80/0.8991	Video, RNN, Intra-frame iterations
	Pan et al. (2020)	Cascade, DAE	Resblock	\mathcal{L}_{pix}	32.13/0.9268	Video, Sharpness Prior, optical flow
	Sun et al. (2015)	DAE	Convolution	\mathcal{L}_{pix}	27.24/0.878	CNN-based approach, MRF
	Gong et al. (2017)	DAE	Fully convolution	$\mathcal{L}_{pix}, \mathcal{L}_{flow}$	28.22/0.894	Estimation of motion flow
	Nah et al. (2017)	MS, GAN	ResBlock	$\mathcal{L}_{per}, \mathcal{L}_{adv}$	29.51/0.912	Multi-scale Net, adversarial loss
	Kupyn et al. (2018)	Conditional GAN	ResBlock	$\mathcal{L}_{per}, \mathcal{L}_{adv}$	26.78/0.848	Conditional GAN-based model
	Zhang et al. (2018a)	U-Net	Recurrent, Residual	\mathcal{L}_{pix}	30.05/0.922	Spatially variant RNN
REDS Nah et al. (2019)	Tao et al. (2018)	U-Net, MS	Recurrent, Dense	\mathcal{L}_{pix}	29.97/0.919	Scale-recurrent Net
	Su et al. (2017)	U-Net	Convolution	\mathcal{L}_{pix}	26.55/0.8066	Video-based deep deblurring Net
	Wang et al. (2019)	Video, DAE	Recurrent, ResBlock	\mathcal{L}_{pix}	34.80/0.9487	Video, Deformable CNN
	Kupyn et al. (2018)	Conditional GAN	ResBlock	$\mathcal{L}_{per}, \mathcal{L}_{adv}$	24.09/0.7482	Conditional GAN-based model
	Nah et al. (2017)	MS, GAN	ResBlock	$\mathcal{L}_{per}, \mathcal{L}_{adv}$	26.16/0.8249	Multi-scale Net, adversarial loss
	Tao et al. (2018)	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 Nah et al. (2019)

Method	Speed (fps)
Su et al. (2017)	1.72
Hyun Kim et al. (2017)	9.24
Nah et al. (2019)	28.8

Table 13 Performance evaluation of different training sets (Martin et al. 2001; Nah et al. 2017; Rim et al. 2020). The image deblurring model is Tao et al. (2018), and the test set is from RealBlur-J dataset Rim et al. (2020). The values are reported in Rim et al. (2020)

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 Nah et al. (2017)	21.12	0.6226
DeblurGAN Kupyn et al. (2018)	19.25	0.5477
SRN Tao et al. (2018)	21.25	0.6233
DeblurGAN-v2 Kupyn et al. (2019)	19.99	0.5865
DMPHN Zhang et al. (2019)	20.98	0.6217

ignore the whole image. We provide a run-time comparison of representative methods in Table 10.

7.2 Video Deblurring

In this section, we compare recent video deblurring approaches on two widely used video deblurring datasets, the DVD Su et al. (2017) and REDS Nah et al. (2019) datasets, see Table 11. Deep auto-encoders Su et al. (2017) are the most commonly used architecture for deep video deblurring methods. Similar to single image deblurring methods, the convolutional layer and ResBlocks He et al. (2016) are the most important components. The recurrent structure Kim et al. (2018) is 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 Zhou et al. (2019) and adversarial loss Zhang et al. (2018) have also been used.

Table 15 Performance evaluation of representative methods for defocus deblurring

Method	PSNR	SSIM
DeepDeblur Nah et al. (2017)	19.78	0.7107
DeblurGAN Kupyn et al. (2018)	19.12	0.6263
SRN Tao et al. (2018)	20.45	0.7557
DeblurGAN-v2 Kupyn et al. (2019)	20.02	0.6896
DMPHN Zhang et al. (2019)	20.12	0.7467

While single image deblurring methods can be applied to videos, they are outperformed by video-based methods that exploit temporal information. Su et al. (2017) 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. (2019) achieves better performance than Su et al. (2017). Zhou et al. (2019) 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. (2020) 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 Sun et al. (2018). A DAE network uses the priors and blurry images to estimate sharp images. Run times of representative methods are shown in Table 12.

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 (Hacohen et al. 2013; Pan et al. 2014). Recently, deep learning solutions have dominated the development of face deblurring by exploiting specific facial characteristics (Jin et al. 2018; Ren et al. 2019; Shen et al. 2018; Xu et al. 2017). Shen et al. (2018) 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. (2018) develop an end-to-end network with a resampling convolution operation to widen the receptive field. Chrysos et al. (2019) propose a two-step architecture, which first restores



Fig. 17 Face deblurring examples. From left to right: input images, results obtained by the methods in Xu et al. (2013), Pan et al. (2014), Nah et al. (2017) and Shen et al. (2018), respectively

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. (2019) 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 (Pan et al. 2014; Xu et al. 2013) and two deep learning based methods (Nah et al. 2017; Shen et al. 2018).

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. (2003) model the text image as a random field to remove blur via blind deconvolution algorithms Fiori et al. (1999). Pan et al. (2014) deblur text images via an L_0 -regularized prior based on intensity and gradient. More recently, deep learning methods, *e.g.* the method by Hradiš et al. (2015) have shown to effectively remove out-of-focus and motion blur. Xu et al. (2017) 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 (Chen et al. 2011; Cho et al. 2012; Cho and Lee 2009; Pan et al. 2014; Xu and Jia 2010; Xu et al. 2013; Zhong et al. 2013) and the deep learning based method in Hradiš et al. (2015), showing that Hradiš et al. (2015) achieves better deblurring results on text images.

8.3 Stereo Image Deblurring

Stereo vision has been widely used to achieve depth perception Godard et al. (2017) and 3D scene understanding Eslami et al. (2016). When mounting a stereo camera on a moving platform, vibration will lead to blur, negatively affecting the subsequent stereo computation Sellent et al.

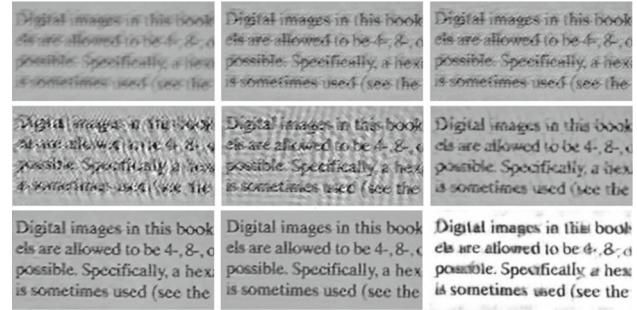


Fig. 18 Text deblurring examples. (From left to right, top to bottom) input image, results from Xu and Jia (2010) Xu et al. (2013), Cho and Lee (2009), Zhong et al. (2013), Chen et al. (2011), Cho et al. (2012), Pan et al. (2014) and Hradiš et al. (2015), respectively

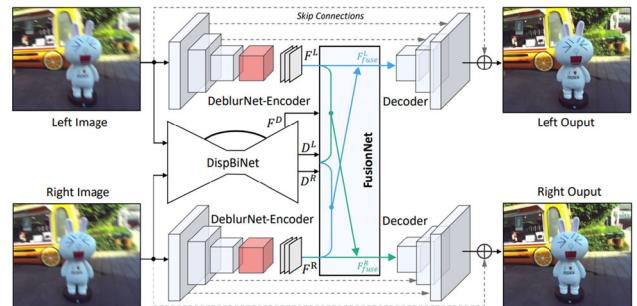


Fig. 19 Stereo image deblurring example Zhou et al. (2019). Two stereo blurry images are fed into deep deblurring networks to generate their corresponding sharp images

(2016). To alleviate this issue, Zhou et al. (2019) 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 (Jin et al. 2018; Purohit et al. 2019), synthesizing high-frame-rate sharp frames Shen et al. (2020), and joint deblurring and super-resolution Zhang et al. (2018).

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 (Nah et al. 2017; Su et al. 2017). 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 (Brooks and Barron 2019; Chen et al. 2018; Zhang et al. 2020). 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 Rim et al. (2020), *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 (Shen et al. 2018; Zhang et al. 2018). 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 Ledig et al. (2017). 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 train-

ing 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 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 (Liu et al. 2018; Pham et al. 2018; Zoph and Le 2017), may also be applied to the deblurring task. In addition, transformers Vaswani et al. (2017) 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. (2018), Nah et al. (2017) and Zhang et al. (2020) 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. (2019), 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. (2019) and Madam Nimisha et al. (2018) 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.

Acknowledgements This research was funded in part by the NSF CAREER Grant #1149783, ARC-Discovery grant projects (DP 19010261 and DP220100800), and a Ford Alliance URP grant.

References

- Abuolaim, A., & Brown, M.S. (2020). Defocus deblurring using dual-pixel data. In European Conference on Computer Vision.
- Aittala, M., & Durand, F. (2018). Burst image deblurring using permutation invariant convolutional neural networks. In European Conference on Computer Vision.
- Aljadaany, R., Pal, D.K., & Savvides, M. (2019). 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.
- Anwar, S., Hayder, Z., & Porikli, F. (2017). Depth estimation and blur removal from a single out-of-focus image. In British Machine Vision Conference.
- Bae, S., & Durand, F. (2007). Defocus magnification. *Computer Graphics Forum*, 26(3), 571–579.
- Bahat, Y., Efrat, N., & Irani, M. (2017). Non-uniform blind deblurring by reblurring. In IEEE Conference on Computer Vision and Pattern Recognition.
- Bigdely, S.A., Zwicker, M., Favaro, P., & Jin, M. (2017). Deep mean-shift priors for image restoration. In Advances in Neural Information Processing Systems, pp. 763–772.
- Blau, Y., & Michaeli, T. (2018). The perception-distortion tradeoff. In IEEE Conference on Computer Vision and Pattern Recognition.
- Boracchi, G., & Foi, A. (2012). Modeling the performance of image restoration from motion blur. *IEEE Transactions on Image Processing*, 21(8), 3502–3517.
- Brooks, T., & Barron, J.T. (2019). Learning to synthesize motion blur. In IEEE Conference on Computer Vision and Pattern Recognition.
- Chakrabarti, A. (2016). A neural approach to blind motion deblurring. In European Conference on Computer Vision.
- Chakrabarti, A., Zickler, T., & Freeman, W.T. (2010). Analyzing spatially-varying blur. In IEEE Conference on Computer Vision and Pattern Recognition.
- Chen, F., & Ma, J. (2009). An empirical identification method of gaussian blur parameter for image deblurring. *IEEE Transactions on Signal Processing*, 57(7), 2467–2478.
- Chen, H., Gu, J., Gallo, O., Liu, M.Y., Veeraraghavan, A., & Kautz, J. (2018). Reblur2deblur: Deblurring videos via self-supervised learning. In IEEE International Conference on Computational Photography.
- Chen, S. J., & Shen, H. L. (2015). Multispectral image out-of-focus deblurring using interchannel correlation. *IEEE Transactions on Image Processing*, 24(11), 4433–4445.
- Chen, X., He, X., Yang, J., & Wu, Q. (2011). An effective document image deblurring algorithm. In IEEE Conference on Computer Vision and Pattern Recognition.
- Cho, H., Wang, J., & Lee, S. (2012). Text image deblurring using text-specific properties. In European Conference on Computer Vision.
- Cho, S., & Lee, S. (2009). Fast motion deblurring. In ACM SIGGRAPH Asia.
- Cho, S., Wang, J., & Lee, S. (2011). Handling outliers in non-blind image deconvolution. In IEEE International Conference on Computer Vision.
- Chrysos, G. G., Favaro, P., & Zafeiriou, S. (2019). Motion deblurring of faces. *International Journal of Computer Vision*, 127(6–7), 801–823.
- Damera-Venkata, N., Kite, T. D., Geisler, W. S., Evans, B. L., & Bovik, A. C. (2000). Image quality assessment based on a degradation model. *IEEE Transactions on Image Processing*, 9(4), 636–650.
- Denton, E.L., Chintala, S., Fergus, R., et al. (2015). Deep generative image models using a laplacian pyramid of adversarial networks. In Advances in Neural Information Processing Systems.
- Dong, J., Roth, S., & Schiele, B. (2020). Deep wiener deconvolution: Wiener meets deep learning for image deblurring. Advances in Neural Information Processing Systems.
- Eigen, D., Puhrsch, C., & Fergus, R. (2014). Depth map prediction from a single image using a multi-scale deep network. In Advances in Neural Information Processing Systems.
- Eslami, S.A., Heess, N., Weber, T., Tassa, Y., Szepesvari, D., Hinton, G.E., et al. (2016). Attend, infer, repeat: Fast scene understanding with generative models. In Advances in Neural Information Processing Systems.
- Fergus, R., Singh, B., Hertzmann, A., Roweis, S.T., & Freeman, W.T. (2006). Removing camera shake from a single photograph. In ACM SIGGRAPH.
- Fiori, S., Uncini, A., & Piazza, F. (1999). Blind deconvolution by modified bussgang algorithm. In The IEEE International Symposium on Circuits and Systems, vol. 3, pp. 1–4.
- Gao, H., Tao, X., Shen, X., & Jia, J. (2019). Dynamic scene deblurring with parameter selective sharing and nested skip connections. In IEEE Conference on Computer Vision and Pattern Recognition.
- Gast, J., Sellent, A., & Roth, S. (2016). Parametric object motion from blur. In IEEE Conference on Computer Vision and Pattern Recognition.
- Godard, C., Mac Aodha, O., & Brostow, G.J. (2017). Unsupervised monocular depth estimation with left-right consistency. In IEEE Conference on Computer Vision and Pattern Recognition.
- Gong, D., Yang, J., Liu, L., Zhang, Y., Reid, I., Shen, C., Van Den Hengel, A., & Shi, Q. (2017). From motion blur to motion flow: a deep learning solution for removing heterogeneous motion blur. In IEEE Conference on Computer Vision and Pattern Recognition.
- Gong, D., Zhang, Z., Shi, Q., van den Hengel, A., Shen, C., & Zhang, Y. (2020). Learning deep gradient descent optimization for image deconvolution. *IEEE Transactions on Neural Networks and Learning Systems*.
- Gu, C., Lu, X., He, Y., & Zhang, C. (2021). Blur removal via blurred-noisy image pair. *IEEE Transactions on Image Processing*, 30(11), 345–359.
- Hacohen, Y., Shechtman, E., & Lischinski, D. (2013). Deblurring by example using dense correspondence. In IEEE International Conference on Computer Vision.
- He, K., Gkioxari, G., Dollár, P., & Girshick, R. (2017). Mask r-cnn. In IEEE International Conference on Computer Vision.
- He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In IEEE Conference on Computer Vision and Pattern Recognition.
- Hirsch, M., Schuler, C.J., Harmeling, S., & Schölkopf, B. (2011). Fast removal of non-uniform camera shake. In IEEE International Conference on Computer Vision.
- Hore, A., & Ziou, D. (2010). Image quality metrics: Psnr vs. ssim. In IEEE International Conference on Pattern Recognition.
- Hoßfeld, T., Heegaard, P. E., Varela, M., & Möller, S. (2016). 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.
- Radíš, M., Kotera, J., Žemcik, P., & Šroubek, F. (2015). Convolutional neural networks for direct text deblurring. In British Machine Vision Conference.
- Hummel, R. A., Kimia, B., & Zucker, S. W. (1987). Deblurring gaussian blur. *Computer Vision, Graphics, and Image Processing*, 38(1), 66–80.
- Hyun Kim, T., Ahn, B., & Mu Lee, K. (2013). Dynamic scene deblurring. In IEEE International Conference on Computer Vision.
- Hyun Kim, T., Mu Lee, K., Scholkopf, B., & Hirsch, M. (2017). Online video deblurring via dynamic temporal blending network. In IEEE International Conference on Computer Vision.
- Isola, P., Zhu, J.Y., Zhou, T., & Efros, A.A. (2017). Image-to-image translation with conditional adversarial networks. In IEEE Conference on Computer Vision and Pattern Recognition.

- Jiang, P., Ling, H., Yu, J., & Peng, J. (2013). Salient region detection by ufo: Uniqueness, focusness and objectness. In IEEE International Conference on Computer Vision.
- Jiang, Z., Zhang, Y., Zou, D., Ren, J., Lv, J., & Liu, Y. (2020). Learning event-based motion deblurring. arXiv preprint [arXiv:2004.05794](https://arxiv.org/abs/2004.05794)
- Jin, M., Hirsch, M., & Favaro, P. (2018). Learning face deblurring fast and wide. In IEEE Conference on Computer Vision and Pattern Recognition Workshop.
- Jin, M., Roth, S., & Favaro, P. (2017). Noise-blind image deblurring. In IEEE Conference on Computer Vision and Pattern Recognition.
- Johnson, J., Alahi, A., & Fei-Fei, L. (2016). Perceptual losses for real-time style transfer and super-resolution. In European Conference on Computer Vision.
- Jolicoeur-Martineau, A. (2018). The relativistic discriminator: a key element missing from standard gan. arXiv preprint [arXiv:1807.00734](https://arxiv.org/abs/1807.00734)
- Kang, S.B. (2007). Automatic removal of chromatic aberration from a single image. In IEEE Conference on Computer Vision and Pattern Recognition.
- Kaufman, A., & Fattal, R. (2020). Deblurring using analysis-synthesis networks pair. arXiv preprint [arXiv:2004.02956](https://arxiv.org/abs/2004.02956)
- Kettunen, M., Härkönen, E., & Lehtinen, J. (2019). E-lpis: robust perceptual image similarity via random transformation ensembles. arXiv preprint [arXiv:1906.03973](https://arxiv.org/abs/1906.03973)
- Kheradmand, A., & Milanfar, P. (2014). A general framework for regularized, similarity-based image restoration. *IEEE Transactions on Image Processing*, 23(12), 5136–5151.
- Kim, T.H., Sajjadi, M.S., Hirsch, M., & Schölkopf, B. (2018). Spatio-temporal transformer network for video restoration. In European Conference on Computer Vision.
- Köhler, R., Hirsch, M., Mohler, B., Schölkopf, B., & Harmeling, S. (2012). Recording and playback of camera shake: Benchmarking blind deconvolution with a real-world database. In European Conference on Computer Vision.
- Krishnan, D., & Fergus, R. (2009). Fast image deconvolution using hyper-laplacian priors. In Advances in Neural Information Processing Systems.
- Kruse, J., Rother, C., & Schmidt, U. (2017). Learning to push the limits of efficient fft-based image deconvolution. In IEEE International Conference on Computer Vision.
- Kupyn, O., Budzan, V., Mykhailych, M., Mishkin, D., & Matas, J. (2018). Deblurgan: Blind motion deblurring using conditional adversarial networks. In IEEE Conference on Computer Vision and Pattern Recognition.
- Kupyn, O., Martyniuk, T., Wu, J., & Wang, Z. (2019). Deblurgan-v2: Deblurring (orders-of-magnitude) faster and better. In IEEE International Conference on Computer Vision.
- Lai, W.S., Huang, J.B., Hu, Z., Ahuja, N., & Yang, M.H. (2016). A comparative study for single image blind deblurring. In IEEE Conference on Computer Vision and Pattern Recognition.
- Le, V., Brandt, J., Lin, Z., Bourdev, L., & Huang, T.S. (2012). Interactive facial feature localization. In European Conference on Computer Vision.
- Ledig, C., Theis, L., Huszár, F., Caballero, J., Cunningham, A., Acosta, A., Aitken, A., Tejani, A., Totz, J., Wang, Z., et al. (2017). Photo-realistic single image super-resolution using a generative adversarial network. In IEEE Conference on Computer Vision and Pattern Recognition.
- Levin, A., Weiss, Y., Durand, F., & Freeman, W.T. (2009). Understanding and evaluating blind deconvolution algorithms. In IEEE Conference on Computer Vision and Pattern Recognition.
- Li, L., Pan, J., Lai, W. S., Gao, C., Sang, N., & Yang, M. H. (2020). Dynamic scene deblurring by depth guided model. *IEEE Transactions on Image Processing*, 29, 5273–5288.
- Li, P., Prieto, L., Mery, D., & Flynn, P. (2018). Face recognition in low quality images: a survey. arXiv preprint [arXiv:1805.11519](https://arxiv.org/abs/1805.11519)
- Li, Y., Tofghi, M., Geng, J., Monga, V., & Eldar, Y. (2019). Deep algorithm unrolling for blind image deblurring. arXiv preprint [arXiv:1902.03493](https://arxiv.org/abs/1902.03493)
- Lin, S., Zhang, J., Pan, J., Jiang, Z., Zou, D., Wang, Y., Chen, J., & Ren, J. (2020). Learning event-driven video deblurring and interpolation. In European Conference on Computer Vision.
- Liu, H., Simonyan, K., & Yang, Y. (2018). Darts: Differentiable architecture search. In International Conference on Learning Representations.
- Liu, L., Liu, B., Huang, H., & Bovik, A. C. (2014). No-reference image quality assessment based on spatial and spectral entropies. *Signal Processing: Image Communication*, 29(8), 856–863.
- Liu, Z., Luo, P., Wang, X., & Tang, X. (2015). Deep learning face attributes in the wild. In IEEE International Conference on Computer Vision.
- Lu, B., Chen, J.C., & Chellappa, R. (2019). Unsupervised domain-specific deblurring via disentangled representations. In IEEE Conference on Computer Vision and Pattern Recognition.
- Lu, Y. (2017). Out-of-focus blur: Image de-blurring. arXiv preprint [arXiv:1710.00620](https://arxiv.org/abs/1710.00620)
- Lumentut, J.S., Kim, T.H., Ramamoorthi, R., & Park, I.K. (2019). Fast and full-resolution light field deblurring using a deep neural network. arXiv preprint [arXiv:1904.00352](https://arxiv.org/abs/1904.00352)
- Madam Nimisha, T., Sunil, K., & Rajagopalan, A. (2018). Unsupervised class-specific deblurring. In European Conference on Computer Vision
- Martin, D., Fowlkes, C., Tal, D., & Malik, J. (2001). 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.
- Masia, B., Corrales, A., Presa, L., & Gutierrez, D. (2011). Coded apertures for defocus deblurring. In Symposium Iberoamericano de Computacion Grafica.
- Michaeli, T., & Irani, M. (2014). Blind deblurring using internal patch recurrence. In European Conference on Computer Vision.
- Mitsa, T., & Varkur, K.L. (1993). 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.
- Mittal, A., Moorthy, A. K., & Bovik, A. C. (2012). No-reference image quality assessment in the spatial domain. *IEEE Transactions on Image Processing*, 21(12), 4695–4708.
- Mittal, A., Soundararajan, R., & Bovik, A. C. (2012). Making a “completely blind” image quality analyzer. *IEEE Signal Processing Letters*, 20(3), 209–212.
- Moorthy, A. K., & Bovik, A. C. (2010). A two-step framework for constructing blind image quality indices. *IEEE Signal Processing Letters*, 17(5), 513–516.
- Moorthy, A. K., & Bovik, A. C. (2011). Blind image quality assessment: From natural scene statistics to perceptual quality. *IEEE Transactions on Image Processing*, 20(12), 3350–3364.
- Mustaniemi, J., Kannala, J., Särkkä, S., Matas, J., & Heikkilä, J. (2019). Gyroscope-aided motion deblurring with deep networks. In IEEE Winter Conference on Applications of Computer Vision.
- Nah, S., Baik, S., Hong, S., Moon, G., Son, S., Timofte, R., & Mu Lee, K. (2019). Ntire 2019 challenge on video deblurring and super-resolution: Dataset and study. In IEEE Conference on Computer Vision and Pattern Recognition Workshop.
- Nah, S., Hyun Kim, T., & Mu Lee, K. (2017). Deep multi-scale convolutional neural network for dynamic scene deblurring. In IEEE Conference on Computer Vision and Pattern Recognition.
- Nah, S., Son, S., & Lee, K.M. (2019). Recurrent neural networks with intra-frame iterations for video deblurring. In IEEE Conference on Computer Vision and Pattern Recognition.
- Nah, S., Son, S., Timofte, R., & Lee, K.M. (2020). Ntire 2020 challenge on image and video deblurring. arXiv preprint [arXiv:2005.01244](https://arxiv.org/abs/2005.01244)

- Nan, Y., Quan, Y., & Ji, H. (2020). Variational-em-based deep learning for noise-blind image deblurring. In IEEE Conference on Computer Vision and Pattern Recognition.
- Niklaus, S., Mai, L., & Liu, F. (2017). Video frame interpolation via adaptive separable convolution. In IEEE International Conference on Computer Vision.
- Nimisha, T.M., Kumar Singh, A., & Rajagopalan, A.N. (2017). Blur-invariant deep learning for blind-deblurring. In IEEE International Conference on Computer Vision.
- Pan, J., Bai, H., & Tang, J. (2020). Cascaded deep video deblurring using temporal sharpness prior. In IEEE Conference on Computer Vision and Pattern Recognition.
- Pan, J., Hu, Z., Su, Z., & Yang, M.H. (2014). Deblurring face images with exemplars. In European Conference on Computer Vision.
- Pan, J., Hu, Z., Su, Z., & Yang, M.H. (2014). Deblurring text images via 10-regularized intensity and gradient prior. In IEEE Conference on Computer Vision and Pattern Recognition.
- Panci, G., Campisi, P., Colonnese, S., & Scarano, G. (2003). Multi-channel blind image deconvolution using the bussgang algorithm: Spatial and multiresolution approaches. *IEEE Transactions on Image Processing*, 12(11), 1324–1337.
- Park, P.D., Kang, D.U., Kim, J., & Chun, S.Y. (2020). Multi-temporal recurrent neural networks for progressive non-uniform single image deblurring with incremental temporal training. In European Conference on Computer Vision.
- Pham, H., Guan, M., Zoph, B., Le, Q., & Dean, J. (2018). Efficient neural architecture search via parameters sharing. In International Conference on Machine Learning.
- Purohit, K., & Rajagopalan, A. (2019). Region-adaptive dense network for efficient motion deblurring. arXiv preprint [arXiv:1903.11394](https://arxiv.org/abs/1903.11394)
- Purohit, K., Shah, A., & Rajagopalan, A. (2019). Bringing alive blurred moments. In IEEE Conference on Computer Vision and Pattern Recognition.
- Ren, D., Zhang, K., Wang, Q., Hu, Q., & Zuo, W. (2019). Neural blind deconvolution using deep priors. arXiv preprint [arXiv:1908.02197](https://arxiv.org/abs/1908.02197)
- Ren, S., He, K., Girshick, R., & Sun, J. (2015). Faster r-cnn: Towards real-time object detection with region proposal networks. In Advances in Neural Information Processing Systems.
- Ren, W., Pan, J., Cao, X., & Yang, M.H. (2017). Video deblurring via semantic segmentation and pixel-wise non-linear kernel. In IEEE International Conference on Computer Vision.
- Ren, W., Yang, J., Deng, S., Wipf, D., Cao, X., & Tong, X. (2019). Face video deblurring using 3d facial priors. In IEEE International Conference on Computer Vision.
- Ren, W., Zhang, J., Ma, L., Pan, J., Cao, X., Zuo, W., Liu, W., & Yang, M.H. (2018). Deep non-blind deconvolution via generalized low-rank approximation. In Advances in Neural Information Processing Systems.
- Rim, J., Lee, H., Won, J., & Cho, S. (2020). Real-world blur dataset for learning and benchmarking deblurring algorithms. In European Conference on Computer Vision.
- Saad, M. A., Bovik, A. C., & Charrier, C. (2012). Blind image quality assessment: A natural scene statistics approach in the dct domain. *IEEE Transactions on Image Processing*, 21(8), 3339–3352.
- Schmidt, U., Rother, C., Nowozin, S., Jancsary, J., & Roth, S. (2013). Discriminative non-blind deblurring. In IEEE Conference on Computer Vision and Pattern Recognition.
- Schuler, C.J., Christopher Burger, H., Harmeling, S., & Schölkopf, B. (2013). A machine learning approach for non-blind image deconvolution. In IEEE Conference on Computer Vision and Pattern Recognition, pp. 1067–1074.
- Schuler, C. J., Hirsch, M., Harmeling, S., & Schölkopf, B. (2015). Learning to deblur. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 38(7), 1439–1451.
- Sellent, A., Rother, C., & Roth, S. (2016). Stereo video deblurring. In European Conference on Computer Vision.
- Sheikh, H. R., & Bovik, A. C. (2006). Image information and visual quality. *IEEE Transactions on Image Processing*, 15(2), 430–444.
- Sheikh, H. R., Bovik, A. C., & De Veciana, G. (2005). An information fidelity criterion for image quality assessment using natural scene statistics. *IEEE Transactions on Image Processing*, 14(12), 2117–2128.
- Shen, W., Bao, W., Zhai, G., Chen, L., Min, X., & Gao, Z. (2020). Blurry video frame interpolation. In IEEE Conference on Computer Vision and Pattern Recognition.
- Shen, Z., Lai, W.S., Xu, T., Kautz, J., & Yang, M.H. (2018). Deep semantic face deblurring. In IEEE Conference on Computer Vision and Pattern Recognition.
- Shen, Z., Lai, W.S., Xu, T., Kautz, J., & Yang, M.H. (2020). Exploiting semantics for face image deblurring. International Journal of Computer Vision pp. 1–18.
- Shen, Z., Wang, W., Lu, X., Shen, J., Ling, H., Xu, T., & Shao, L. (2019). Human-aware motion deblurring. In IEEE International Conference on Computer Vision.
- Shi, J., Xu, L., & Jia, J. (2014). Discriminative blur detection features. In IEEE Conference on Computer Vision and Pattern Recognition.
- Sim, H., & Kim, M. (2019). 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.
- Sim, T., Baker, S., & Bsas, M. (2002). The cmu pose, illumination, and expression (PIE) database. In IEEE International Conference on Automatic Face Gesture Recognition.
- Simonyan, K., & Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. arXiv preprint [arXiv:1409.1556](https://arxiv.org/abs/1409.1556)
- Son, C. H., & Park, H. M. (2011). A pair of noisy/blurry patches-based psf estimation and channel-dependent deblurring. *IEEE Transactions on Image Processing*, 57(4), 1791–1799.
- Su, S., Delbracio, M., Wang, J., Sapiro, G., Heidrich, W., & Wang, O. (2017). Deep video deblurring for hand-held cameras. In IEEE Conference on Computer Vision and Pattern Recognition.
- Suin, M., Purohit, K., & Rajagopalan, A. (2020). Spatially-attentive patch-hierarchical network for adaptive motion deblurring. arXiv preprint [arXiv:2004.05343](https://arxiv.org/abs/2004.05343)
- Sun, D., Yang, X., Liu, M.Y., & Kautz, J. (2018). PWC-Net: Cnns for optical flow using pyramid, warping, and cost volume. In IEEE Conference on Computer Vision and Pattern Recognition.
- Sun, J., Cao, W., Xu, Z., & Ponce, J. (2015). Learning a convolutional neural network for non-uniform motion blur removal. In IEEE Conference on Computer Vision and Pattern Recognition.
- Sun, L., Cho, S., Wang, J., & Hays, J. (2013). Edge-based blur kernel estimation using patch priors. In IEEE International Conference on Computational Photography.
- Sun, L., & Hays, J. (2012). Super-resolution from internet-scale scene matching. In IEEE International Conference on Computational Photography.
- Sun, T., Peng, Y., & Heidrich, W. (2017). Revisiting cross-channel information transfer for chromatic aberration correction. In IEEE International Conference on Computer Vision, pp. 3248–3256.
- Szeliski, R. (2010). Computer vision: algorithms and applications. Springer Science & Business Media.
- Tang, C., Zhu, X., Liu, X., Wang, L., & Zomaya, A. (2019). Defusionnet: Defocus blur detection via recurrently fusing and refining multi-scale deep features. In IEEE Conference on Computer Vision and Pattern Recognition.
- Tao, X., Gao, H., Shen, X., Wang, J., & Jia, J. (2018). Scale-recurrent network for deep image deblurring. In IEEE Conference on Computer Vision and Pattern Recognition.
- Vairy, M., & Venkatesh, Y. V. (1995). Deblurring gaussian blur using a wavelet array transform. *Pattern Recognition*, 28(7), 965–976.

- Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L., & Polosukhin, I. (2017) Attention is all you need. arXiv preprint [arXiv:1706.03762](https://arxiv.org/abs/1706.03762)
- Wang, X., Chan, K.C., Yu, K., Dong, C., & Change Loy, C. (2019). EDVR: Video restoration with enhanced deformable convolutional networks. In IEEE Conference on Computer Vision and Pattern Recognition Workshop.
- Wang, Z., & Bovik, A.C. (2002). A universal image quality index. *IEEE Signal Processing Letters*, 9(3), 81–84.
- Wang, Z., Bovik, A.C., Sheikh, H.R., & Simoncelli, E.P. (2004). Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4), 600–612.
- Wang, Z., Simoncelli, E.P., & Bovik, A.C. (2003). Multiscale structural similarity for image quality assessment. In The Asilomar Conference on Signals, Systems, and Computers.
- Whyte, O., Sivic, J., & Zisserman, A. (2014). Deblurring shaken and partially saturated images. *International Journal of Computer Vision*, 110(2), 185–201.
- Whyte, O., Sivic, J., Zisserman, A., & Ponce, J. (2012). Non-uniform deblurring for shaken images. *International Journal of Computer Vision*, 98(2), 168–186.
- Wieschollek, P., Hirsch, M., Scholkopf, B., & Lensch, H. (2017). Learning blind motion deblurring. In IEEE International Conference on Computer Vision.
- Xia, F., Wang, P., Chen, L.C., & Yuille, A.L. (2016). Zoom better to see clearer: Human and object parsing with hierarchical auto-zoom net. In European Conference on Computer Vision.
- Xu, L., & Jia, J. (2010). Two-phase kernel estimation for robust motion deblurring. In European Conference on Computer Vision.
- Xu, L., Ren, J.S., Liu, C., & Jia, J. (2014). Deep convolutional neural network for image deconvolution. In Advances in Neural Information Processing Systems.
- Xu, L., Tao, X., & Jia, J. (2014). Inverse kernels for fast spatial deconvolution. In European Conference on Computer Vision.
- Xu, L., Zheng, S., & Jia, J. (2013). Unnatural 10 sparse representation for natural image deblurring. In IEEE Conference on Computer Vision and Pattern Recognition.
- Xu, X., Pan, J., Zhang, Y.J., & Yang, M.H. (2017). Motion blur kernel estimation via deep learning. *IEEE Transactions on Image Processing*, 27(1), 194–205.
- Xu, X., Sun, D., Pan, J., Zhang, Y., Pfister, H., & Yang, M.H. (2017). Learning to super-resolve blurry face and text images. In IEEE International Conference on Computer Vision.
- Yasarla, R., Perazzi, F., & Patel, V.M. (2019). Deblurring face images using uncertainty guided multi-stream semantic networks. arXiv preprint [arXiv:1907.13106](https://arxiv.org/abs/1907.13106)
- Ye, P., Kumar, J., Kang, L., & Doermann, D. (2012). Unsupervised feature learning framework for no-reference image quality assessment. In IEEE Conference on Computer Vision and Pattern Recognition.
- Zhang, H., Dai, Y., Li, H., & Koniusz, P. (2019). Deep stacked hierarchical multi-patch network for image deblurring. In IEEE Conference on Computer Vision and Pattern Recognition.
- Zhang, J., Pan, J., Lai, W.S., Lau, R.W., & Yang, M.H. (2017). Learning fully convolutional networks for iterative non-blind deconvolution. In IEEE Conference on Computer Vision and Pattern Recognition.
- Zhang, J., Pan, J., Ren, J., Song, Y., Bao, L., Lau, R.W., & Yang, M.H. (2018). Dynamic scene deblurring using spatially variant recurrent neural networks. In IEEE Conference on Computer Vision and Pattern Recognition.
- Zhang, K., Luo, W., Zhong, Y., Ma, L., Liu, W., & Li, H. (2018). Adversarial spatio-temporal learning for video deblurring. *IEEE Transactions on Image Processing*, 28(1), 291–301.
- Zhang, K., Luo, W., Zhong, Y., Stenger, B., Ma, L., Liu, W., & Li, H. (2020). Deblurring by realistic blurring. In IEEE Conference on Computer Vision and Pattern Recognition.
- Zhang, K., Van Gool, L., & Timofte, R. (2020). Deep unfolding network for image super-resolution. In IEEE Conference on Computer Vision and Pattern Recognition.
- Zhang, K., Zuo, W., Gu, S., & Zhang, L. (2017). Learning deep cnn denoiser prior for image restoration. In IEEE Conference on Computer Vision and Pattern Recognition.
- Zhang, K., Zuo, W., & Zhang, L. (2018). Learning a single convolutional super-resolution network for multiple degradations. In IEEE Conference on Computer Vision and Pattern Recognition.
- Zhang, K., Zuo, W., & Zhang, L. (2019). Deep plug-and-play super-resolution for arbitrary blur kernels. In IEEE Conference on Computer Vision and Pattern Recognition.
- Zhang, R., Isola, P., Efros, A.A., Shechtman, E., & Wang, O. (2018). The unreasonable effectiveness of deep features as a perceptual metric. In IEEE Conference on Computer Vision and Pattern Recognition.
- Zhang, W., & Cham, W.K. (2009). Single image focus editing. In IEEE International Conference on Computer Vision Workshop.
- Zhang, X., Dong, H., Hu, Z., Lai, W.S., Wang, F., & Yang, M.H. (2018). Gated fusion network for joint image deblurring and super-resolution. arXiv preprint [arXiv:1807.10806](https://arxiv.org/abs/1807.10806)
- Zhao, W., Zheng, B., Lin, Q., & Lu, H. (2019). Enhancing diversity of defocus blur detectors via cross-ensemble network. In IEEE Conference on Computer Vision and Pattern Recognition.
- Zhong, L., Cho, S., Metaxas, D., Paris, S., & Wang, J. (2013). Handling noise in single image deblurring using directional filters. In IEEE Conference on Computer Vision and Pattern Recognition.
- Zhong, Z., Gao, Y., YinQiang, Z., & Bo, Z. (2020). Efficient spatio-temporal recurrent neural network for video deblurring. In European Conference on Computer Vision.
- Zhou, S., Zhang, J., Pan, J., Xie, H., Zuo, W., & Ren, J. (2019). Spatio-temporal filter adaptive network for video deblurring. In IEEE International Conference on Computer Vision.
- Zhou, S., Zhang, J., Zuo, W., Xie, H., Pan, J., & Ren, J.S. (2019). Davanet: Stereo deblurring with view aggregation. In IEEE Conference on Computer Vision and Pattern Recognition.
- Zhu, J.Y., Park, T., Isola, P., & Efros, A.A. (2017). Unpaired image-to-image translation using cycle-consistent adversarial networks. In IEEE International Conference on Computer Vision.
- Zoph, B., & Le, Q.V. (2017). Neural architecture search with reinforcement learning. In International Conference on Learning Representations.
- Zoran, D., & Weiss, Y. (2011). From learning models of natural image patches to whole image restoration. In IEEE International Conference on Computer Vision.

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.