NTIRE 2020 Challenge on Image and Video Deblurring

Seungjun Nah   Sanghyun Son   Radu Timofte   Kyoung Mu Lee   Yu Tseng
Yu-Syuan Xu   Cheng-Ming Chiang   Yi-Min Tsai   Stephan Brehm   Sebastian Scherer
Dejia Xu   Yihao Chu   Qingyan Sun   Jiaqin Jiang   Lunhao Duan   Jian Yao
Kuldeep Purpohit   Maitreya Suin   A.N. Rajagopalan   Yuichi Ito   Hrishikesh P S
Densen Puthussery   Akhil K A   Jiji C V   Guisik Kim   Deepa P L   Zhiwei Xiong
Jie Huang   Dong Liu   Sangmin Kim   Hyungjoon Nam   Jisu Kim   Jechang Jeong
Shihua Huang   Yuchen Fan   Jiahui Yu   Haichao Yu   Thomas S. Huang   Ya Zhou
Xin Li   Sen Liu   Zhibo Chen   Saikat Dutta   Sourya Dipta Das   Shivam Garg
Daniel Sprague   Bhrij Patel   Thomas Huck

Abstract

Motion blur is one of the most common degradation artifacts in dynamic scene photography. This paper reviews the NTIRE 2020 Challenge on Image and Video Deblurring. In this challenge, we present the evaluation results from 3 competition tracks as well as the proposed solutions. Track 1 aims to develop single-image deblurring methods focusing on restoration quality. On Track 2, the image deblurring methods are executed on a mobile platform to find the balance of the running speed and the restoration accuracy. Track 3 targets developing video deblurring methods that exploit the temporal relation between input frames. In each competition, there were 163, 135, and 102 registered participants and in the final testing phase, 9, 4, and 7 teams competed. The winning methods demonstrate the state-of-the-art performance on image and video deblurring tasks.

1. Introduction

As smartphones are becoming the most popular type of cameras in the world, snapshot photographs are prevailing. Due to the dynamic nature of the hand-held devices and free-moving subjects, motion blurs are commonly witnessed on images and videos. Computer vision literature has studied post-processing methods to remove blurs from photography by parametrizing camera motion [16, 22, 80] or generic motion [19, 31, 41, 34, 35, 36].

Modern machine learning based computer vision methods employ large-scale datasets to train their models. For image and video deblurring tasks, GOPRO [50], DVD [66], WILD [55] datasets were proposed as ways to synthesize blurry images by mimicking camera imaging pipeline from high-frame-rate videos. Recent dynamic scene deblurring methods train on such datasets to develop image [50, 72, 92, 33] and video [81, 37, 38, 51, 32, 8] deblurring methods.

However, the early datasets for deblurring lacked in the quality of reference images and were limited in the realism of blur. With the NTIRE 2019 workshop, a new improved dataset, REDS [49] was proposed by providing longer sensor read-out time, using a measured camera-response function and interpolating frames, *etc*. REDS dataset was employed for NTIRE 2019 Video Deblurring [52] and Super-Resolution [53] Challenges.

Succeeding the previous year, NTIRE 2020 Challenge on Image and Video Deblurring presents 3 competition tracks. In track 1, single image deblurring methods are submitted and competed on desktop environments, focusing on the image restoration accuracy in terms of PSNR and SSIM. In track 2, similarly to track 1, single image deblurring methods are submitted but deployed on a mobile device. Considering the practical application environment, the image restoration quality as well as the running time is evaluated together. Track 3 exploits temporal information, continuing the NTIRE 2019 Video Deblurring Challenge.

This challenge is one of the NTIRE 2020 associated challenges: deblurring, nonhomogeneous dehazing [4], perceptual extreme super-resolution [89], video quality mapping [13], real image denoising [2], real-world super-resolution [46], spectral reconstruction from RGB image [5] and demoireing [86].

2. Related Works

The REDS dataset is designed for non-uniform blind deblurring. As all the methods submitted to the 3 tracks use deep neural networks, we describe the related works. Also, we describe the previous studies on neural network deployment on mobile devices.

2.1. Single Image Deblurring

Traditional energy optimization methods [19, 31, 41, 34, 35] jointly estimated the blur kernel and the latent sharp image from a blurry image. Early deep learning methods tried to obtain the blur kernel from neural networksby estimating local motion field [69, 15] for later latent image estimation. From the advent of the image deblurring datasets [50, 66], end-to-end learning methods were presented by estimating the deblurred outputs directly [50, 72, 92] without kernel estimation. From the initial coarse-to-fine architecture [50] whose parameters were scale-specific, scale-recurrent networks [72] and selective sharing scheme [14] has been proposed. Spatially varying operation was studied by using spatial RNN in [92] While the previous methods behave in class-agnostic manner, as face and human bodies are often being the main subject of photographs, class-specific deblurring methods were proposed [63, 60]. They employ semantic information or 3D facial priors to better reconstruct target class objects. On another flow of studies, several approaches [33, 57] were proposed to extract multiple sharp images from a single blurry image. They are motivated by the fact that a blurry images have long exposure that could be an accumulation of many sharp frames with short exposure.

2.2. Video Deblurring

Video Deblurring methods exploit the temporal relation between the input frames in various manners. DBN [66] stacks 5 consecutive frames in channels and a CNN learns to aggregate the information between the frames. The frames are aligned by optical flow before being fed into the network. Recurrent neural networks are used to pass the information from the past frames to the next ones [81, 37, 38, 51]. RDN [81] and OVD [37] performs additional connections to the hidden state steps to better propagate useful information. STTN [38] was proposed on top of DBN and OVD to exploit long-range spatial correspondence information. IFI-RNN [51] investigates into reusing parameters to make hidden states more useful. Recently proposed STFAN [94] proposes filter adaptive convolution(FAC) to apply element-wise convolution kernels. As the convolutional kernel is acquired from the feature value, the operation is spatially varying.

At the NTIRE 2019 Video Deblurring Challenge [52], EDVR [77] was the winning solution. PCD alignment module is devised to handle large motion and TSA fusion module is proposed to utilize the spatio-temporal attention.

2.3. Neural Network for Mobile Deployment

While deep neural network methods have brought significant success in computer vision, most of them require heavy computation and large memory footprint. For practical usage such as mobile deployment, lightweight and efficient model architectures are required [26, 27]. Therefore, several methods have been developed to compress pretrained neural networks while keeping their performances.

For example, network quantization aims to reduce the number of bits for representing weight parameters and intermediate features. With a degree of performance loss, binary [11, 59] or ternary [42] quantization significantly reduce model complexity and memory usage. To utilize efficient bit-shift operation, network parameters can be represented as powers of two [93].

On the other hand, several studies tried to limit the number of parameters by pruning [17, 18, 21] or using a set of parameters with indexed representations [9]. Recent pruning methods consider convolution operation structure [43, 47, 65]. Also, as convolutional filters are multi-dimensional, the weights were approximated by decomposition techniques [30, 65, 91, 75].

On another point of view, several methods are aiming to design efficient network architectures. MobileNet V1 [24], V2 [62], ThunderNet [58] adopt efficient layer compositions to build lightweight models in carefully handcrafted manners. Furthermore, MnasNet [70], MobileNet V3 [23], EfficientNet [71] employ neural architecture search to find low-latency models while keeping high accuracy. Despite the success in architecture efficiency, most of the networks compression methods have been mostly focusing on high-level tasks such as image recognition and object detection.

As general image restoration requires 8-bit depth at minimum, it is a nontrivial issue to apply the previous compression techniques to design lightweight models. Thus, [48] utilizes a local quantization technique in image super-resolution task by binarizing layers in the residual blocks only. Besides, there were attempts to apply efficient designs of MobileNets [24, 62] for super-resolution and deblurring. In [3], efficient residual block design was presented while [40] applied the depthwise separable convolution.Recently, [44] proposed to linearly decompose the convolutional filters and learn the basis of filters with optional basis sharing between layers. These methods effectively reduces the model computation while preserving the restoration accuracy in a similar level.

PIRM 2018 [25] and AIM 2019 [88] organized challenges for image super-resolution and image enhancement on smartphones and for constrained super-resolution, respectively. We refer the reader to [26, 27] for an overview

and benchmark of Android smartphones and their capability to run deep learning models.

3. The Challenge

We hosted the NTIRE 2020 Image and Video Deblurring Challenge in order to promote developing the state-of-the-art algorithms for image and video deblurring. Following the NTIRE 2019 Challenge on Video Deblurring and Super-Resolution, we use the REDS dataset [49] to measure the performance of the results.

3.1. Tracks

In this challenge, we divide the competition into 3 tracks: (1) Image Deblurring (2) Image Deblurring on Mobile Devices (3) Video deblurring.

Track 1: Image Deblurring aims to develop single-image deblurring methods without limiting the computational resources.

Track 2: Image Deblurring on Mobile Devices goes beyond simply developing well-performing single-image deblurring methods. To encourage steps towards more practical solutions, the running time is measured as well as the restoration accuracy. The challenge participants are requested to submit TensorFlow Lite models so that the running speed is measured by the organizers.

We use Google Pixel 4 (Android 10) as the platform to deploy the deblurring models. The processor is Qualcomm snapdragon 855 which supports GPU and DSP acceleration. 32-bit floating point models typically choose to run on GPU while the 8-bit quantized models further accelerate on DSP via Android NNAPI.

Track 3: Video Deblurring targets developing video deblurring methods that exploit temporal relation between the video frames. The winner of the NTIRE 2019 challenge, EDVR [77] learns to align the input feature and performs deblurring afterwards. In this challenge, there were several attempts to use additional modules on top of EDVR.

3.2. Evaluation

The competition consists of development and testing phases. During the development phase, the registered participants train their method and could get feedback from online server to validate the solution format. Meanwhile, they can get local feedback directly from the validation data. In the testing phase, the participants submit their results as well as the source code, trained model, and the fact sheets describing their solution. The reproducibility is checked by the organizers.

The results are basically evaluated by the conventional accuracy metrics: PSNR and SSIM [79]. For track 2, we use a score term that favors both fast and accurate methods. Compared with our baseline model, we add a relative fps

score to the PSNR of the restoration results. Finally, our score functions is

$$\text{score} = \text{PSNR} + \frac{1}{2} \log_2 \frac{\text{FPS}}{\text{FPS}_{baseline}}, \qquad (1)$$

where our baseline is a simple residual network with 4 blocks and 64 channels [45]. $\text{FPS}_{baseline}$ is 8.23. The methods faster than the baseline gets higher score than its PSNR while the slower are penalized. The gains from acceleration is evaluated in log scale to prevent extremely fast methods without meaningful processing Also, we set the maximum scorable fps to be 45. However, no submission score was plateaued by the fps limit.

4. Challenge Results

Each challenge track had 163, 135, and 102 registered participants. In the final testing phase, total 9, 5, 8 results were submitted. The teams submitted the deblurred frames as well as the source code and the trained models.

Table 1, 2, 3 each summarizes the result of the corresponding competition track. All the proposed methods on desktop use deep neural networks with GPU acceleration. Mobile accelerators are shown in Table 2.

Baseline methods We present the baseline method results to compare with the participants' methods. For track 1, we present the result of Nah *et al.* [50] that is trained with the REDS dataset. L1 loss is used to train the model for 200 epochs with batch size 16. The learning rate was set to 10^{-4} and halved at 100th, 120th, and 140th epoch. For track 2, we present a EDSR [45]-like architecture without upscaling module. 4 ResBlocks are used with 64 channels. For track 3, we compare the results with EDVR [77], the winner of NTIRE 2019 Video Deblurring Challenge.

Architectures and Main ideas There were several attempts to use multi-scale information in different perspectives. UniA team used atrous convolution [85] and uses multi-scale aggregation for video deblurring MTKur, Wangwang, CVML, VIDAR, Reboot, Duke Data Science chose encoder-decoder or U-net style architecture while Vermilion used SRN [72]. In contrast, IPCV_IITM, CET_CVLab, Neuro_avengers adopted DMPHN [87] without scaling. On the other hand, OIerM used fractal architecture to fuse multi-depth information.

For video deblurring, EMI_VR, IMCL-PROMOTION modified EDVR [77]. UIUC-IFP modified their previous WDVR [12] by frame concatenation.

Several teams used loss other than L1 or L2 loss: adversarial loss (CET Deblurring Team, SG), WAE-MMD loss (CVML), perceptual loss(IMCL-PROMOTION, SG).

Restoration fidelity UniA Team, MTKur are the winners of NTIRE 2020 challenge in track 1 and 2. In track 3, the submitted results did not improve from the NTIRE 2019 winner, EDVR.

Team	Method	Track 1: Image Deblurring			
		PSNR	SSIM	FPS	GPU
UniA Team	Wide Atrous Block (ensemble ×4)	34.44	0.9412	1.43	Tesla V100
OIerM	Attentive Fractal Network	34.20	0.9392	1.16	RTX 2080 Ti
MTKur	DRU-prelu (ensemble ×3)	33.35	0.9283	0.83	RTX 2080 Ti
Wangwang	Two-stage EdgeDeblurNet	33.07	0.9242	0.46	GTX 1080 Ti
IPCV_IITM	DMPHN + Region-Adaptive Network (ensemble ×8)	33.03	0.9242	0.56	RTX 2080 Ti
Vermilion	Simplified SRN (ensemble ×3)	30.04	0.8616	0.36	GTX 1080 (eGPU)
CET_CVLAB	Stack-DMPHN	29.78	0.8629	0.50	Quadro K6000
CVML	Wasserstein Autoencoder	28.10	0.8097	33.3	RTX 2080 SUPER
CET Deblurring Team	DoubleGAN	26.58	0.7492	2.04	GTX 1080
baseline	Multi-scale CNN [50][1]	32.90	0.9207	1.12	RTX 2080 Ti
baseline	no processing	26.13	0.7749	-	-

Table 1: Single image deblurring results on the REDS test data.

Team	Method	Track 2: Image Deblurring (Mobile)				
		PSNR	SSIM	FPS[2]	Final score	Accelerator
MTKur	DRU-relu-compressed	32.07	0.9024	17.6	32.62	NNAPI
	DRU-prelu	32.95	0.9239	5.1	32.60	GPU
VIDAR	Transformed fusion U-Net	30.20	0.8735	6.0	29.97	GPU
Reboot	Light-weight Attention Network	31.38	0.8960	1.0	29.87	GPU
OIerM	Attentive Fractal Network	28.33	0.8079	0.9	26.71	CPU
baseline	4 ResBlocks	28.46	0.8218	8.23	28.46	GPU
baseline	no processing	26.13	0.7749	-	-	-

Table 2: Single image deblurring results on the REDS test data from Google Pixel 4.

Team	Method	Track 3: Video Deblurring			
		PSNR	SSIM	FPS	GPU
EMI_VR	PAFU (ensemble ×4)	36.93	0.9649	0.14	Tesla V100
UIUC-IFP	WDVR+ (ensemble ×8)	35.58	0.9504	0.07	GTX 1080 Ti
IMCL-PROMOTION	PROMOTION	35.42	0.9519	0.94	GTX 1080 Ti
UniA Team	Dual-Stage Multi-Level Feature Aggregation	34.68	0.9442	0.18	Tesla V100
Neuro_avengers	DMPHN + GridNet	31.85	0.9074	0.40	Titan X
	DMPHN	31.43	0.8949	0.97	Titan X
SG	Multi-Loss Optimization	29.44	0.8526	0.48	GTX Titan Black
Duke Data Science	Encoder-Decoder	26.88	0.8051	0.24	Tesla V100
NTIRE 2019 winner	EDVR [77]	36.94[3]	0.9656	0.26	Titan Xp
baseline	no processing	26.13	0.7749	-	-

Table 3: Video deblurring results on the REDS test data.

5. Challenge Methods and Teams

We describe the submitted solution details in this section.

5.1. MTKur team - Track 1, 2

MTKur team is the winner of Track 2. Following the guidelines in [10], MTKur team develops Dense Residual U-Net (DRU) architecture by applying several modifications to U-Net [61]. First, they replace concatenation operation in all skip connections with addition operation except the last global connection. Second, they replace the single convolution operation with a dense residual block to improve deblurring quality. Third, considering the mobile deployment, TransposeConv operations are replaced with ResizeBilinear operations as they have poor latency on Pixel 4. The overall DRU architecture is shown in Figure 1.

Based on DRU, MTKur proposes 2 variations: DRU-prelu and DRU-relu-compressed depending on the purpose. DRU-prelu targets on better restoration quality by using PReLU activation [20] after all convolutional layers except the convolutional layers before ResizeBilinear and the last convolutional layer. In contrast, DRU-relu-compressed uses ReLU activation aiming higher throughput of the model. Also, current Tensorflow Lite kernel generates erroneous outputs of quantized PReLU layer.

To DRU-relu-compressed, a series of network compression techniques provided in MediaTek's NeuroPilot toolkit [1] are applied including pruning and quantization after basic training. First, iterative pruning scheme [76] exploited. The model repeats to prune by 5% MAC reduction criterion and retrains to achieve the original PSNR quality. After the iteration, the pruned model achieves 20% MAC reduction without PSNR drop in the REDS validation dataset. Then, the network is further optimized

---

[1]trained on REDS dataset without adversarial loss.

[2]256 × 256 resolution is used as processing on mobile devices is slow.

[3]In NTIRE 2020, boundaries are included in the evaluation.

with quantization-aware training of NeuroPilot toolkit [1] to minimize PSNR drop.

To obtain better results, geometric self-ensemble(x3) [73, 45] is applied to DRU-prelu in Track 1. In Track 2, ensemble was not used for better running time.

The proposed models are trained with 256×256 sized patches of batch size 16. L_1 loss is applied at learning rate 2×10^{-4}, exponential decaying by rate 0.98 with 5000 decay steps. The input and target images are normalized to range [0, 1]. The training before compression takes 10 days on a single RTX 2080 Ti. Pruning and quantization takes 6 and 2 days, respectively.
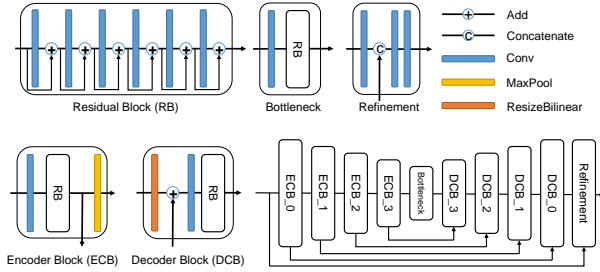
Figure 1: MTKur team: Dense Residual U-Net

5.2. UniA team - Track 1, 3

UniA team proposes an image deblurring and a video deblurring method [6] for track 1 and 3, respectively. In track 1, they propose to use atrous convolution to increase the receptive field instead of downsampling input multiple times to prevent loss of information. From the experiments, receptive fields with missing pixels inside were not beneficial. Thus, Wide Atrous Block is designed where parallel atrous convolutions with different dilation rates are used. The features are concatenated to be used in the next layer. The model architecture is shown in 2. To stabilize the training, the convolutional features are scaled by learnable parameters and added by a constant before the activation. LeakyReLU activation is used except the last layer which uses a linear activation.

The model is trained with extensive data augmentation including brightness, contrast, hue augmentation. Patch size is set to 320×320. VGG, edge similarity, adversarial loss were investigated but didn't bring improvements in terms of PSNR an d SSIM. Geometric self-ensemble is applied to improve performance [73, 45].

The video deblurring method performs post-processing to the results obtaine from the image deblurring method. The features from the target and the neighboring frame are fused in multiple scales. From downsampled frames, independent convolutional features are concatenated and fused by residual blocks. The coarse features are upsampled and
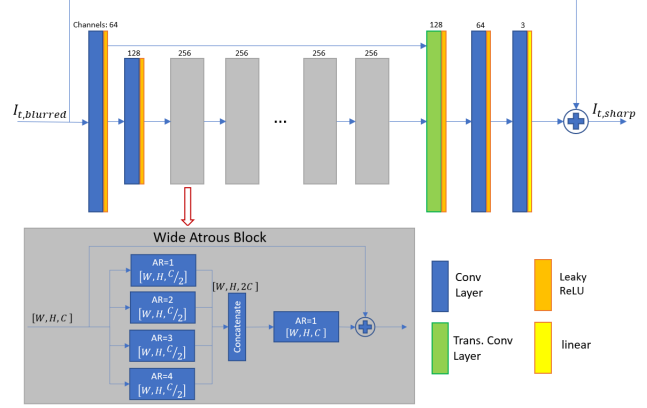

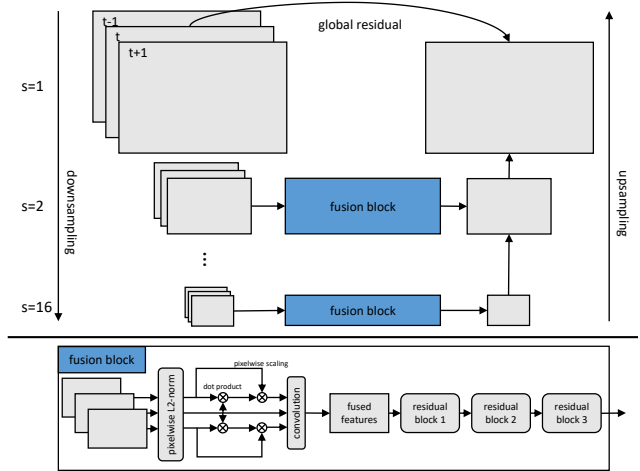Figure 2: UniA team (Track 1): Atrous Convolutional Network

Figure 3: UniA team (Track 3): Dual-Stage Multi-Level Feature Aggregation

added to the higher resolution features. Figure 3 shows the video deblurring architecture.

5.3. OIerM team - Track 1, 2

OIerM team proposes an Attentive Fractal Network (AFN) [83]. They construct the Attentive Fractal Block via progressive feature fusion and channel-wise attention guidance. Then AFB is stacked in a fractal way inspired by FBL-SS [84] such that a higher-level AFB_s is constructed with the lower-level AFB_{s-1}, recursively with self-similarity, as shown in Figure 4. Shortcuts and residual connections at different scale effectively resolve the vanishing gradients and help the network learn more key features. The progressive fusion of intermediate features let the network handle rich information.

For track 1 and 2, n, m, and s in Figure 4 are set to 4 and 2, respectively. The models are trained with batch size 8 and 16 for 50 and 200 epochs. The channel size is 128.
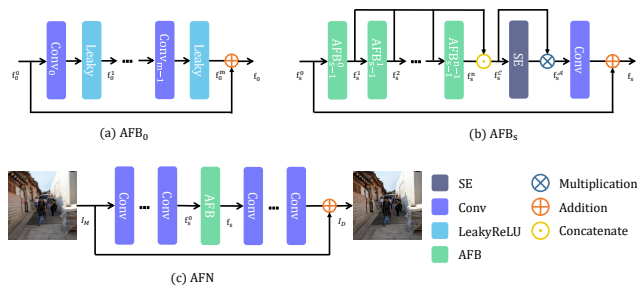
Figure 4: OIerM team: Attentive Fractal Network

5.4. Wangwang team - Track 1

Wangwang team proposes a solution, EdgeDeblurNet, based on the DeblurNet module of DAVANet [95]. As the edges and smooth area are differently blurred, they employ the edge map as extra available information to guide deblurring. To better emphasize the main object edges than the blur trajectories, spatial and channel attention gate [82] is adopted. Also, SFT-ResBlock [78] is used to fuse the edge information effectively. Finally, inspired by the traditional iterative optimization methods, a two-stage deblurring method is proposed by cascading the EdgeDeblurNet.

The second stage model is trained after the first stage parameters are trained and fixed. The model is trained with 256×256 patches of batch size 16 by Adam [39] optimizer for 400 epochs. The learning rate is initialized as 10^{-4} and decayed at predetermined steps. The overall architecture is shown in Figure 5.
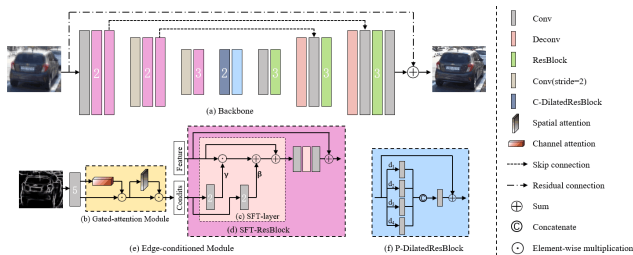

Figure 5: Wangwang team: Two-stage Edge-Conditioned Network

5.5. IPCV_IITM team - Track 1

IPCV_IITM team buids a two-stage network based on DMPHN [87] and [67]. The first stage is a 3-level DMPHN with cross-attention modules similar to [67] where each pixel can gather global information. The second stage is a densely connected encoder-decoder structure [56]. Before the last layer, The multi-scale context aggregation is performed through pooling and upsampling. Geometric self-ensemble [73, 45] is applied to further improve results. Figure 6 shows the overall architecture.
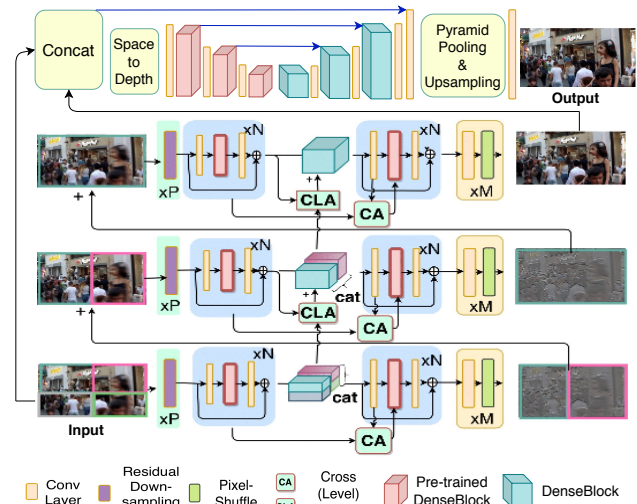

Figure 6: IPCV_IITM team: Region-Adaptive Patch-hierarchical Network

5.6. Vermilion team - Track 1

Vermilion team uses a simplified Scale-Recurrent Networks [72] in 4 scale levels. To make SRN simpler, the recurrent connections are removed. On each level, a series of ResBlocks are used instead of U-Net based structure. The final deblurred result is obtained by an ensemble of 3 results.

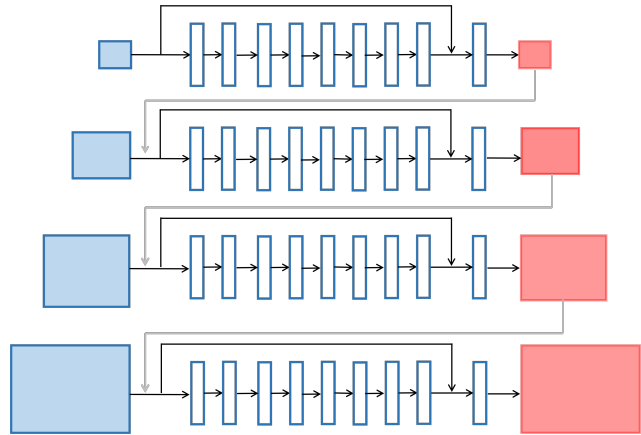The model is trained with 256×256 patches of batch size 4. The learning rate was set to 10^{-4}.

Figure 7: Vermilion team: Simplified SRN architecture

5.7. CET_CVLab team - Track 1

CET_CVLab team uses Stack-DMPHN [87] with 5 levels. The upsampling method is replaced by depth-to-space operation. The model is trained with MSE loss with Adam optimizer. The learning rate is initialized with 10^{-4} and halved at every 100 epochs.

5.8. CVML team - Track 1

CVML team uses a Wasserstein autoencoder [74] for single image deblurring. The latent space is represented as spatial tensor instead of 1D vector. In addition, an advanced feature based architecture is designed to deliver rich feature information to the latent space. There are three tree-structured fusion modules for encoder and decoder, respectively. To train the model, WAE-maximum mean discrepancy (WAE-MMD) loss, reconstruction loss, and perceptual loss is applied to the 240×240 patches. The overall architecture is shown in Figure 8.
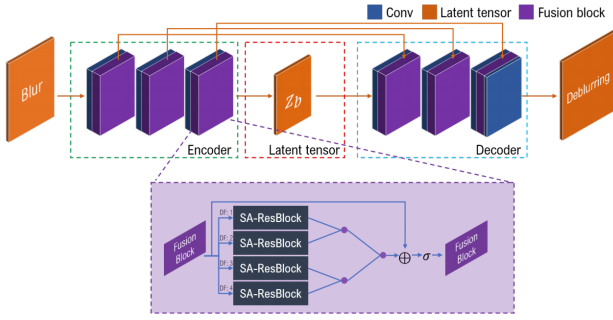

Figure 8: CVML team: Wasserstein Autoencoder

5.9. CET Deblurring team - Track 1

CET Deblurring team proposes a double generative adversarial network that consists of two generators and discriminators. The generator G_1 and G_2 performs image deblurring and the inverse operation. As G_2 is trained with VGG-19 [64] loss to recover blurry image, G_1 learns to deblur the original and the re-blurred. The two discriminators try to classify the output from G_1 and the ground-truth with different objectives. D_f focuses on the edges by looking at the gray LoG features while D_I refers to the color intensity in HSI format from Gaussian blurred images. The overall architecture is shown in Figure 9.
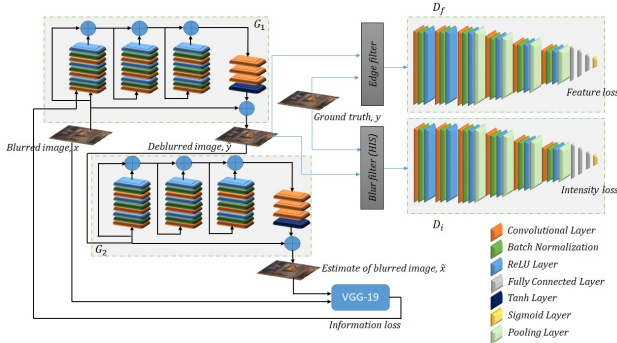

Figure 9: CET_Deblurring team: Double GAN

5.10. VIDAR team - Track 2

VIDAR team proposes a Transformed fusion U-Net. In the connections between the corresponding encoder and decoder, the encoder features go through a transform layer to get useful information. Also, the decoder features from different scales are concatenated together. Finally, different dilated convolution layers are used that are similar to atrous pyramid pooling to estimate different blur motion.

To train the model, L1 loss, SSIM loss, and multi-scale loss are applied. The training patch size is 256×256 and the batch size is 16. Adam optimizer is used with learning rate decaying from 2.2×10^{-5} by power 0.997. Training stops when the training loss does not notably decreases.

5.11. Reboot team - Track 2

Reboot team proposes a light-weight attention network from their proposed building blocks. TUDA (Total Up-Down Attention) block is the high-level block that contains other blocks: UAB and UDA. UAB (U-net based Attention Block) resembles U-net like architecture that replaces convolutions with UDA. UDA (Up-Down Attention block) is the basic block that operates like ResBlock but in a down-sampled scale for efficiency. The overall architecture is shown in Figure 10.

To train the model, 96×96 patches are used with batch size 32. Adam optimizer is used with learning rate beginning from 10^{-4} that is halved every 50 epochs.
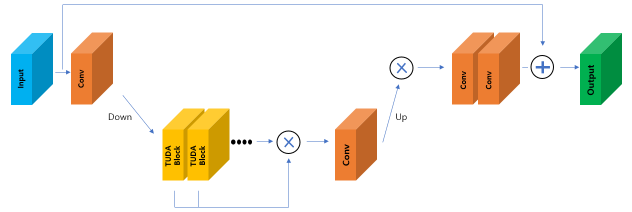

Figure 10: Reboot team: Light-weight Attention Network

5.12. EMI_VR team - Track 3

EMI_VR team proposes a framework, PAFU, that tries to refine the result of 2 serial stack of EDVR [77]. PAFU architecture is similar to EDVR but has different modules. First, PAFU uses non-local spatial attention before PreDeblur operation. After the PreDeblur process, a series of AFU (Alignment, Fusion, and Update) modules perform progressly on the extracted feature. AFU module performs FAD (Feature Align with Deformable convolution) and TCA (Temporal Channel Attention) to fuse aligned features. Figure 11 shows the overall architecture of PAFU.

To train the model, both the training and validation data are used except 4 training videos that are selected by the authors. At test time, geometric self-ensemble [73, 45] using flips are used.
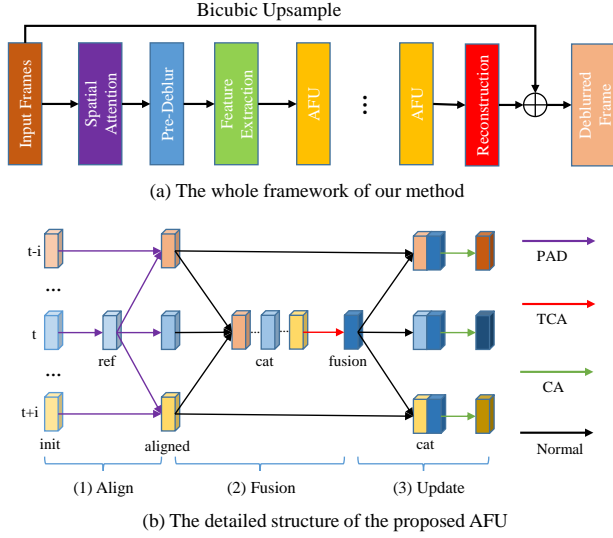
(a) The whole framework of our method

(b) The detailed structure of the proposed AFU

Figure 11: EMI_VR team: PAFU

5.13. UIUC-IFP team - Track 3

UIUC-IFP team proposes WDVR+ based on WDVR [12]. In the 2D convolutional WDVR, the input frames were stacked along the channel axis. WDVR+ extends WDVR by motion compensation in feature space via template matching. Given a patch of the center frame as a template, the most correlated patches in the neighbor frames are searched. Such patches are stacked as input for the deblurring model. The template matching process is jointly learned with the deep networks. At test time, geometric self-ensemble [73, 45] is used.

5.14. IMCL-PROMOTION team - Track 3

IMCL-PROMOTION team proposes a framework named PROMOTION [96] based on the pretrained EDVR [77] architecture.. When deblurring models are not properly trained, they may expect the inputs to be always blurry and worsen several sharp inputs by unnecessary operation. To let the model distinguish blurry and sharp input, blur reasoning vector is designed to represent the blur degree of each frame. Also, to better handle non-uniform nature of motion blur, extra features are extracted. From the input frames, image contrast and gradient maps are extracted. To encode the motion information, optical flow of the center frame is calculated by FlowNet 2.0 [28]. As calculating optial flow for all the frames is heavy, for the neighbor frames, frame difference is used instead. Those maps are fed into 3D convolutional networks to encode the heterogeneous information. The extracted feature is used at the reconstruction module. The overall architecture is shown in Figure 12. To train the model, Charbonnier loss [7] and the perceptual loss [90].
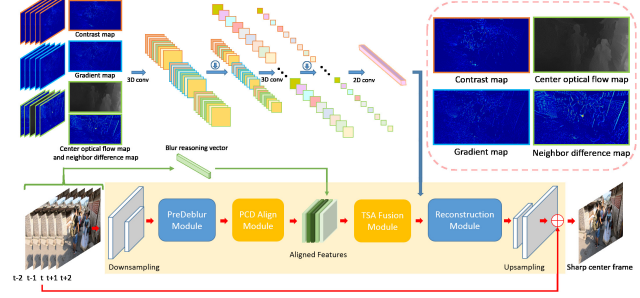

Figure 12: IMCL-PROMOTION team: PROMOTION

5.15. Neuro_avengers team

Neuro_avengers team proposes 2 methods by modifying DMPHN [87]. In the bottom level, a set of subsequent frames are concatenated while from the higher levels recover the target frame only. They propose the second by refining the result of method 1 by cascading GridNet [54]. GridNet takes the 2 neighboring frames warped by optical flow [68] as well.

5.16. SG team - Track 3

SG team trains their model with multiple loss functions: L1, VGG-16, SSIM, and adversarial loss. Each convolution is followed by batch normalization [29] and PReLU except the first two layers without BN. The overall architecture is shown in Figure 13. To handle temporal relation, multiple frames were concatenated as input to the network.
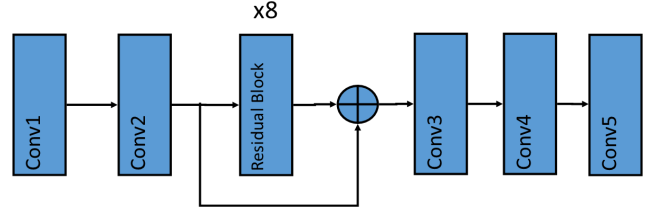

Figure 13: SG team: Multi-Loss Optimization

5.17. Duke Data Science team - Track 3

Duke Data Science team implemented a simple encoder-decoder network based on Su *et al*. [66]. The input to the network is a stack of 5 consecutive video frames separated in the color channel. There are 3 independent encoder-decoder modules that handle a single color channel.

Acknowledgments

A. Teams and affiliations

NTIRE 2020 team

Title: NTIRE 2020 Challenge on Image and Video Deblurring
Members: Seungjun Nah[1] (seungjun.nah@gmail.com), Sanghyun Son[1], Radu Timofte[2], Kyoung Mu Lee[1]
Affiliations:
[1] Department of ECE, ASRI, SNU, Korea
[2] Computer Vision Lab, ETH Zurich, Switzerland

MTKur

Title 1: Dense Residual U-Net for Single Image Deblurring
Title 2: Toward Efficient Dense Residual U-Net for Single Image Deblurring
Members: Cheng-Ming Chiang (jimmy.chiang@mediatek.com), Yu Tseng, Yu-Syuan Xu, Yi-Min Tsai
Affiliations:
MediaTek Inc.

UniA Team

Title 1: Atrous Convolutional Block for Image Deblurring
Title 2: Dual-Stage Multi-Level Feature Aggregation for Video Deblurring
Members: Stephan Brehm (stephan.brehm@informatik.uni-augsburg.de), Sebastian Scherer
Affiliations:
University of Augsburg, Chair for Multimedia Computing and Computer Vision Lab, Germany

OIerM

Title: Attentive Fractal Network
Members: Dejia Xu[1] (dejia@pku.edu.cn), Yihao Chu[2], Qingyan Sun[3]
Affiliations:
[1] Peking University, China
[2] Beijing University of Posts and Telecommunications, China
[3] Beijing Jiaotong University, China

Wangwang

Title: Two-stage Edge-Conditioned Network for Deep Image Deblurring
Members: Jiaqin Jiang (jiangjiaqin@whu.edu.cn), Lunhao Duan, Jian Yao
Affiliations:
Wuhan University, China

IPCV_IITM

Title: Region-Adaptive Patch-hierarchical Network for Single Image Deblurring
Members: Kuldeep Purohit (kuldeeppurohit3@gmail.com), Maitreya Suin, A.N. Rajagopalan
Affiliations:
Indian Institute of Technology Madras, India

Vermilion

Title: Simplified SRN
Members: Yuichi Ito (wataridori2010@gmail.com)
Affiliations:
Vermilion

CET_CVLab

Title: V-Stacked Deep CNN for Single Image Deblurring
Members: Hrishikesh P S (hrishikeshps@cet.ac.in), Densen Puthussery, Akhil K A, Jiji C V
Affiliations:
College of Engineering Trivandrum

CVML

Title: Image Deblurring using Wasserstein Autoencoder
Members: Guisik Kim (specialre@naver.com)
Affiliations:
CVML, Chung-Ang University, Korea

CET Deblurring Team

Title: Blind Image Deblurring using Double Generative Adversarial Network
Members: Deepa P L (deepa.pl@mbcet.ac.in), Jiji C V
Affiliations:
APJ Abdul Kalam Technological University, India

VIDAR

Title: Transformed fusion U-Net
Members: Zhiwei Xiong (zwxiong@ustc.edu.cn), Jie Huang, Dong Liu
Affiliations:
University of Science and Technology of China, China

Reboot

Title: Light-weight Attention Network for Image Deblurring on Smartphone

Members: *Sangmin Kim (ksmh1652@gmail.com)*, Hyungjoon Nam, Jisu Kim, Jechang Jeong
Affiliations:
Image Communication Signal Processing Laboratory, Hanyang University, Korea

EMI_VR

Title: Progressive Alignment, Fusion and Update for Video Restoration
Members: *Shihua Huang (shihuahuang95@gmail.com)*
Affiliations:
Southern University of Science and Technology, China

UIUC-IFP

Title: WDVR+: Motion Compensation via Feature Template Matching
Members: *Yuchen Fan (yc0624@gmail.com)*, Jiahui Yu, Haichao Yu, Thomas S. Huang
Affiliations:
University of Illinois at Urbana-Champaign

IMCL-PROMOTION

Title: Prior-enlightened and Motion-robust Video Deblurring
Members: *Ya Zhou (zhouya@mail.ustc.edu.cn)*, Xin Li, Sen Liu, Zhibo Chen
Affiliations:
CAS Key Laboratory of Technology in Geo-Spatial Information Processing and Application System, University of Science and Technology of China

Neuro_avengers

Title: Deep Multi-patch Hierarchical Network for Video Deblurring
Members: *Saikat Dutta[1] (saikat.dutta779@gmail.com)*, Sourya Dipta Das
Affiliations:
[1] IIT Madra
[2] Jadavpur University

SG

Title: Multi-Loss Optimization for Video Deblurring
Members: *Shivam Garg (shivgarg@live.com)*
Affiliations:
University of Texas at Austin, USA

Duke Data Science

Title: Encoder-Decoder
Members: *Daniel Sprague (dys9@duke.edu)*, Bhrij Patel, Thomas Huck
Affiliations:
Duke University Computer Science Department

References

[1] MediaTek NeuroPilot SDK. https://neuropilot.mediatek.com/. 4, 5

[2] Abdelrahman Abdelhamed, Mahmoud Afifi, Radu Timofte, Michael Brown, et al. NTIRE 2020 challenge on real image denoising: Dataset, methods and results. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, June 2020. 1

[3] Namhyuk Ahn, Byungkon Kang, and Kyung-Ah Sohn. Fast, accurate, and lightweight super-resolution with cascading residual network. In *The European Conference on Computer Vision (ECCV)*, September 2018. 2

[4] Codruta O. Ancuti, Cosmin Ancuti, Florin-Alexandru Vasluianu, Radu Timofte, et al. NTIRE 2020 challenge on nonhomogeneous dehazing. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, June 2020. 1

[5] Boaz Arad, Radu Timofte, Yi-Tun Lin, Graham Finlayson, Ohad Ben-Shahar, et al. NTIRE 2020 challenge on spectral reconstruction from an rgb image. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, June 2020. 1

[6] Stephan Brehm, Sebastian Scherer, and Rainer Lienhart. High-resolution dual-stage multi-level feature aggregation for single image and video deblurring. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, June 2020. 5

[7] Pierre Charbonnier, Laure Blanc-Feraud, Gilles Aubert, and Michel Barlaud. Two deterministic half-quadratic regularization algorithms for computed imaging. In *Proceedings of 1st International Conference on Image Processing*, volume 2, pages 168–172. IEEE, 1994. 8

[8] Huaijin Chen, Jinwei Gu, Orazio Gallo, Ming-Yu Liu, Ashok Veeraraghavan, and Jan Kautz. Reblur2deblur: Deblurring videos via self-supervised learning. In *2018 IEEE International Conference on Computational Photography (ICCP)*, pages 1–9. IEEE, 2018. 1

[9] Wenlin Chen, James Wilson, Stephen Tyree, Kilian Weinberger, and Yixin Chen. Compressing neural networks with the hashing trick. In *International conference on machine learning*, pages 2285–2294, 2015. 2

[10] Cheng-Ming Chiang, Yu Tseng, Yu-Syuan Xu, Hsien-Kai Kuo, Yi-Min Tsai, Guan-Yu Chen, Kaon-Sin Tan, Wei-Ting Wang, Yu-Chieh Lin, Shou-Yao Roy Tseng, Wei-Shiang Lin, Chia-Lin Yu, BY Shen, Kloze Kao, Chia-Ming Cheng, and Hung-Jen Chen. Deploying image deblurring across mobile devices: A perspective of quality and latency. In *The*

IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops, June 2020. 4

[11] Matthieu Courbariaux, Itay Hubara, Daniel Soudry, Ran El-Yaniv, and Yoshua Bengio. Binarized neural networks: Training deep neural networks with weights and activations constrained to+ 1 or-1. *arXiv preprint arXiv:1602.02830*, 2016. 2

[12] Yuchen Fan, Jiahui Yu, Ding Liu, and Thomas S. Huang. An empirical investigation of efficient spatio-temporal modeling in video restoration. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, June 2019. 3, 8

[13] Dario Fuoli, Zhiwu Huang, Martin Danelljan, Radu Timofte, et al. NTIRE 2020 challenge on video quality mapping: Methods and results. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, June 2020. 1

[14] Hongyun Gao, Xin Tao, Xiaoyong Shen, and Jiaya Jia. Dynamic scene deblurring with parameter selective sharing and nested skip connections. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019. 2

[15] Dong Gong, Jie Yang, Lingqiao Liu, Yanning Zhang, Ian Reid, Chunhua Shen, Anton van den Hengel, and Qinfeng Shi. From motion blur to motion flow: A deep learning solution for removing heterogeneous motion blur. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017. 2

[16] Ankit Gupta, Neel Joshi, C Lawrence Zitnick, Michael Cohen, and Brian Curless. Single image deblurring using motion density functions. In *The European Conference on Computer Vision (ECCV)*, pages 171–184. Springer, 2010. 1

[17] Song Han, Huizi Mao, and William J Dally. Deep compression: Compressing deep neural networks with pruning, trained quantization and huffman coding. *arXiv preprint arXiv:1510.00149*, 2015. 2

[18] Song Han, Jeff Pool, John Tran, and William Dally. Learning both weights and connections for efficient neural network. In C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett, editors, *Advances in Neural Information Processing Systems 28*, pages 1135–1143. Curran Associates, Inc., 2015. 2

[19] Stefan Harmeling, Hirsch Michael, and Bernhard Schölkopf. Space-variant single-image blind deconvolution for removing camera shake. In J. D. Lafferty, C. K. I. Williams, J. Shawe-Taylor, R. S. Zemel, and A. Culotta, editors, *Advances in Neural Information Processing Systems 23*, pages 829–837. Curran Associates, Inc., 2010. 1, 2

[20] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *The IEEE International Conference on Computer Vision (ICCV)*, December 2015. 4

[21] Yihui He, Xiangyu Zhang, and Jian Sun. Channel pruning for accelerating very deep neural networks. In *The IEEE International Conference on Computer Vision (ICCV)*, Oct 2017. 2

[22] Michael Hirsch, Christian J Schuler, Stefan Harmeling, and Bernhard Schölkopf. Fast removal of non-uniform camera sshake. In *IEEE International Conference on Computer Vision (ICCV)*, pages 463–470. IEEE, 2011. 1

[23] Andrew Howard, Mark Sandler, Grace Chu, Liang-Chieh Chen, Bo Chen, Mingxing Tan, Weijun Wang, Yukun Zhu, Ruoming Pang, Vijay Vasudevan, Quoc V. Le, and Hartwig Adam. Searching for mobilenetv3. In *The IEEE International Conference on Computer Vision (ICCV)*, October 2019. 2

[24] Andrew G Howard, Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, Marco Andreetto, and Hartwig Adam. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861*, 2017. 2

[25] Andrey Ignatov and Radu Timofte. PIRM challenge on perceptual image enhancement on smartphones: Report. In *The European Conference on Computer Vision (ECCV) Workshops*, September 2018. 2

[26] Andrey Ignatov, Radu Timofte, William Chou, Ke Wang, Max Wu, Tim Hartley, and Luc Van Gool. Ai benchmark: Running deep neural networks on android smartphones. In *The European Conference on Computer Vision (ECCV) Workshops*, September 2018. 2

[27] Andrey Ignatov, Radu Timofte, Andrei Kulik, Seungsoo Yang, Ke Wang, Felix Baum, Max Wu, Lirong Xu, and Luc Van Gool. Ai benchmark: All about deep learning on smartphones in 2019. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV) Workshops*, pages 3617–3635. IEEE, 2019. 2

[28] Eddy Ilg, Nikolaus Mayer, Tonmoy Saikia, Margret Keuper, Alexey Dosovitskiy, and Thomas Brox. Flownet 2.0: Evolution of optical flow estimation with deep networks. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017. 8

[29] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv preprint arXiv:1502.03167*, 2015. 8

[30] Max Jaderberg, Andrea Vedaldi, and Andrew Zisserman. Speeding up convolutional neural networks with low rank expansions. *arXiv preprint arXiv:1405.3866*, 2014. 2

[31] Hui Ji and Kang Wang. A two-stage approach to blind spatially-varying motion deblurring. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 73–80. IEEE, 2012. 1, 2

[32] Meiguang Jin, Zhe Hu, and Paolo Favaro. Learning to extract flawless slow motion from blurry videos. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019. 1

[33] Meiguang Jin, Givi Meishvili, and Paolo Favaro. Learning to extract a video sequence from a single motion-blurred image. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018. 1, 2

[34] Tae Hyun Kim, Byeongjoo Ahn, and Kyoung Mu Lee. Dynamic scene deblurring. In *The IEEE International Conference on Computer Vision (ICCV)*, December 2013. 1, 2

[35] Tae Hyun Kim and Kyoung Mu Lee. Segmentation-free dynamic scene deblurring. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2014. 1, 2

[36] Tae Hyun Kim and Kyoung Mu Lee. Generalized video deblurring for dynamic scenes. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015. 1

[37] Tae Hyun Kim, Kyoung Mu Lee, Bernhard Scholkopf, and Michael Hirsch. Online video deblurring via dynamic temporal blending network. In *The IEEE International Conference on Computer Vision (ICCV)*, October 2017. 1, 2

[38] Tae Hyun Kim, Mehdi S. M. Sajjadi, Michael Hirsch, and Bernhard Scholkopf. Spatio-temporal transformer network for video restoration. In *The European Conference on Computer Vision (ECCV)*, September 2018. 1, 2

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

[40] Orest Kupyn, Tetiana Martyniuk, Junru Wu, and Zhangyang Wang. Deblurgan-v2: Deblurring (orders-of-magnitude) faster and better. In *The IEEE International Conference on Computer Vision (ICCV)*, October 2019. 2

[41] Anat Levin. Blind motion deblurring using image statistics. In *Advances in Neural Information Processing Systems*, pages 841–848, 2007. 1, 2

[42] Fengfu Li, Bo Zhang, and Bin Liu. Ternary weight networks. *arXiv preprint arXiv:1605.04711*, 2016. 2

[43] Hao Li, Asim Kadav, Igor Durdanovic, Hanan Samet, and Hans Peter Graf. Pruning filters for efficient convnets. *arXiv preprint arXiv:1608.08710*, 2016. 2

[44] Yawei Li, Shuhang Gu, Luc Van Gool, and Radu Timofte. Learning filter basis for convolutional neural network compression. In *The IEEE International Conference on Computer Vision (ICCV)*, October 2019. 2

[45] Bee Lim, Sanghyun Son, Heewon Kim, Seungjun Nah, and Kyoung Mu Lee. Enhanced deep residual networks for single image super-resolution. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, July 2017. 3, 5, 6, 7, 8

[46] Andreas Lugmayr, Martin Danelljan, Radu Timofte, et al. NTIRE 2020 challenge on real-world image super-resolution: Methods and results. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, June 2020. 1

[47] Jian-Hao Luo, Jianxin Wu, and Weiyao Lin. Thinet: A filter level pruning method for deep neural network compression. In *The IEEE International Conference on Computer Vision (ICCV)*, Oct 2017. 2

[48] Yinglan Ma, Hongyu Xiong, Zhe Hu, and Lizhuang Ma. Efficient super resolution using binarized neural network. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, June 2019. 2

[49] Seungjun Nah, Sungyong Baik, Seokil Hong, Gyeongsik Moon, Sanghyun Son, Radu Timofte, and Kyoung Mu Lee. NTIRE 2019 challenges on video deblurring and super-resolution: Dataset and study. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, June 2019. 1, 3

[50] Seungjun Nah, Tae Hyun Kim, and Kyoung Mu Lee. Deep multi-scale convolutional neural network for dynamic scene deblurring. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017. 1, 2, 3, 4

[51] Seungjun Nah, Sanghyun Son, and Kyoung Mu Lee. Recurrent neural networks with intra-frame iterations for video deblurring. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019. 1, 2

[52] Seungjun Nah, Radu Timofte, Sungyong Baik, Seokil Hong, Gyeongsik Moon, Sanghyun Son, and Kyoung Mu Lee. NTIRE 2019 challenge on video deblurring: Methods and results. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, June 2019. 1, 2

[53] Seungjun Nah, Radu Timofte, Shuhang Gu, Sungyong Baik, Seokil Hong, Gyeongsik Moon, Sanghyun Son, and Kyoung Mu Lee. NTIRE 2019 challenge on video super-resolution: Methods and results. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, June 2019. 1

[54] Simon Niklaus and Feng Liu. Context-aware synthesis for video frame interpolation. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018. 8

[55] Mehdi Noroozi, Paramanand Chandramouli, and Paolo Favaro. Motion deblurring in the wild. In *German conference on pattern recognition*, pages 65–77. Springer, 2017. 1

[56] Kuldeep Purohit and AN Rajagopalan. Region-adaptive dense network for efficient motion deblurring. *arXiv preprint arXiv:1903.11394*, 2019. 6

[57] Kuldeep Purohit, Anshul Shah, and A. N. Rajagopalan. Bringing alive blurred moments. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019. 2

[58] Zheng Qin, Zeming Li, Zhaoning Zhang, Yiping Bao, Gang Yu, Yuxing Peng, and Jian Sun. Thundernet: Towards real-time generic object detection on mobile devices. In *The IEEE International Conference on Computer Vision (ICCV)*, October 2019. 2

[59] Mohammad Rastegari, Vicente Ordonez, Joseph Redmon, and Ali Farhadi. Xnor-net: Imagenet classification using binary convolutional neural networks. In *The European Conference on Computer Vision (ECCV)*, pages 525–542. Springer, 2016. 2

[60] Wenqi Ren, Jiaolong Yang, Senyou Deng, David Wipf, Xiaochun Cao, and Xin Tong. Face video deblurring using 3d facial priors. In *The IEEE International Conference on Computer Vision (ICCV)*, October 2019. 2

[61] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015. 4

[62] Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. Mobilenetv2: Inverted residuals and linear bottlenecks. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018. 2

[63] Ziyi Shen, Wenguan Wang, Xiankai Lu, Jianbing Shen, Haibin Ling, Tingfa Xu, and Ling Shao. Human-aware mo-

tion deblurring. In *The IEEE International Conference on Computer Vision (ICCV)*, October 2019. 2

[64] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. 7

[65] Sanghyun Son, Seungjun Nah, and Kyoung Mu Lee. Clustering convolutional kernels to compress deep neural networks. In *The European Conference on Computer Vision (ECCV)*, September 2018. 2

[66] Shuochen Su, Mauricio Delbracio, Jue Wang, Guillermo Sapiro, Wolfgang Heidrich, and Oliver Wang. Deep video deblurring for hand-held cameras. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017. 1, 2, 8

[67] Maitreya Suin, Kuldeep Purohit, and A. N. Rajagopalan. Spatially-attentive patch-hierarchical network for adaptive motion deblurring. *arXiv preprint arXiv:2004.05343*, 2020. 6

[68] Deqing Sun, Xiaodong Yang, Ming-Yu Liu, and Jan Kautz. Pwc-net: Cnns for optical flow using pyramid, warping, and cost volume. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018. 8

[69] Jian Sun, Wenfei Cao, Zongben Xu, and Jean Ponce. Learning a convolutional neural network for non-uniform motion blur removal. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015. 2

[70] Mingxing Tan, Bo Chen, Ruoming Pang, Vijay Vasudevan, Mark Sandler, Andrew Howard, and Quoc V. Le. Mnasnet: Platform-aware neural architecture search for mobile. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019. 2

[71] Mingxing Tan and Quoc V Le. Efficientnet: Rethinking model scaling for convolutional neural networks. *arXiv preprint arXiv:1905.11946*, 2019. 2

[72] Xin Tao, Hongyun Gao, Xiaoyong Shen, Jue Wang, and Jiaya Jia. Scale-recurrent network for deep image deblurring. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018. 1, 2, 3, 6

[73] Radu Timofte, Rasmus Rothe, and Luc Van Gool. Seven ways to improve example-based single image super resolution. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016. 5, 6, 7, 8

[74] Ilya Tolstikhin, Olivier Bousquet, Sylvain Gelly, and Bernhard Schoelkopf. Wasserstein auto-encoders. *arXiv preprint arXiv:1711.01558*, 2017. 7

[75] Min Wang, Baoyuan Liu, and Hassan Foroosh. Factorized convolutional neural networks. In *The IEEE International Conference on Computer Vision (ICCV) Workshops*, Oct 2017. 2

[76] Wei-Ting Wang, Han-Lin Li, Wei-Shiang Lin, Cheng-Ming Chiang, and Yi-Min Tsai. Architecture-aware network pruning for vision quality applications. In *2019 IEEE International Conference on Image Processing (ICIP)*, pages 2701–2705. IEEE, 2019. 4

[77] Xintao Wang, Kelvin C.K. Chan, Ke Yu, Chao Dong, and Chen Change Loy. Edvr: Video restoration with enhanced

deformable convolutional networks. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, June 2019. 2, 3, 4, 7, 8

[78] Xintao Wang, Ke Yu, Chao Dong, and Chen Change Loy. Recovering realistic texture in image super-resolution by deep spatial feature transform. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018. 6

[79] Zhou Wang, Alan C Bovik, Hamid R Sheikh, Eero P Simoncelli, et al. Image quality assessment: from error visibility to structural similarity. *The IEEE Transactions on Image Processing (TIP)*, 13(4):600–612, 2004. 3

[80] Oliver Whyte, Josef Sivic, Andrew Zisserman, and Jean Ponce. Non-uniform deblurring for shaken images. *International journal of computer vision*, 98(2):168–186, 2012. 1

[81] Patrick Wieschollek, Michael Hirsch, Bernhard Scholkopf, and Hendrik P. A. Lensch. Learning blind motion deblurring. In *The IEEE International Conference on Computer Vision (ICCV)*, October 2017. 1, 2

[82] Sanghyun Woo, Jongchan Park, Joon-Young Lee, and In So Kweon. Cbam: Convolutional block attention module. In *The European Conference on Computer Vision (ECCV)*, September 2018. 6

[83] Dejia Xu, Yihao Chu, and Qingyan Sun. Moiré pattern removal via attentive fractal network. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, June 2020. 5

[84] Wenhan Yang, Shiqi Wang, Dejia Xu, Xiaodong Wang, and Jiaying Liu. Towards scale-free rain streak removal via self-supervised fractal band learning. February 2020. 5

[85] Fisher Yu and Vladlen Koltun. Multi-scale context aggregation by dilated convolutions. *arXiv preprint arXiv:1511.07122*, 2015. 3

[86] Shanxin Yuan, Radu Timofte, Ales Leonardis, Gregory Slabaugh, et al. NTIRE 2020 challenge on image demoireing: Methods and results. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, June 2020. 1

[87] Hongguang Zhang, Yuchao Dai, Hongdong Li, and Piotr Koniusz. Deep stacked hierarchical multi-patch network for image deblurring. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019. 3, 6, 8

[88] Kai Zhang, Shuhang Gu, Radu Timofte, et al. AIM 2019 challenge on constrained super-resolution: Methods and results. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV) Workshops*, pages 3565–3574. IEEE, 2019. 2

[89] Kai Zhang, Shuhang Gu, Radu Timofte, et al. NTIRE 2020 challenge on perceptual extreme super-resolution: Methods and results. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, June 2020. 1

[90] Richard Zhang, Phillip Isola, Alexei A. Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018. 8

[91] Xiangyu Zhang, Jianhua Zou, Kaiming He, and Jian Sun. Accelerating very deep convolutional networks for classification and detection. *IEEE transactions on pattern analysis and machine intelligence*, 38(10):1943–1955, 2015. 2

[92] Yulun Zhang, Yapeng Tian, Yu Kong, Bineng Zhong, and Yun Fu. Residual dense network for image super-resolution. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018. 1, 2

[93] Aojun Zhou, Anbang Yao, Yiwen Guo, Lin Xu, and Yurong Chen. Incremental network quantization: Towards lossless cnns with low-precision weights. *arXiv preprint arXiv:1702.03044*, 2017. 2

[94] Shangchen Zhou, Jiawei Zhang, Jinshan Pan, Haozhe Xie, Wangmeng Zuo, and Jimmy Ren. Spatio-temporal filter adaptive network for video deblurring. In *The IEEE International Conference on Computer Vision (ICCV)*, October 2019. 2

[95] Shangchen Zhou, Jiawei Zhang, Wangmeng Zuo, Haozhe Xie, Jinshan Pan, and Jimmy S. Ren. Davanet: Stereo deblurring with view aggregation. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019. 6

[96] Ya Zhou, Jianfeng Xu, Kazuyuki Tasakab, Zhibo Chen, and Weiping Li. Prior-enlightened and motion-robust video deblurring. *arXiv preprint arXiv:2003.11209*, 2020. 8