DOUBLE CLOSED-LOOP NETWORK FOR IMAGE DEBLURRING

Yiming Liu[1, 2+], *Yanni Zhang*[1,+], *Qiang Li*[1], *Jun Kong*[1,3], *Miao Qi*[3, *] and *Jianzhong Wang*[1,*]

[1] College of Information Science and Technology, Northeast Normal University, China
[2] College of Information Science and Engineering, Hunan Normal University, China
[3] Institute for Intelligent Elderly Care, Changchun Humanities and Sciences College, China

ABSTRACT

In this paper, a deep learning network with double closed-loop structure is introduced to tackle the image deblurring problem. The first closed-loop in our model is composed of two networks which learn a pair of opposite mappings between the blurry and sharp images. By this way, the solution spaces of possible functions that map a blurry image to its sharp counterpart can be effectively reduced. Furthermore, the first closed-loop also helps our model to deal with the unpaired samples in the training set. The second closed-loop in the proposed approach employed a self-supervision mechanism to constrain the features of intermedia layers in the network, so that the detailed information of sharp images can be well exploited. Through combining the two closed-loops together, our model can address the limitations of existing methods and improve the deblurring performance. Extensive experiments on both benchmark and real-world datasets show that the proposed network achieves state-of-the-art performance. The code will be released in: https://github.com/LiQiang0307/DCLNet.

Index Terms—Image deblurring, closed-loop structure, cycle consistency, solution space

1. INTRODUCTION

Blur is an adverse factor which would degrade the quality of images and deteriorate the performance of many real-world applications. Therefore, image deblurring which aims to recover a sharp image from its blurry source is an active topic in computer vision research community. However, due to the image blur can be caused by various factors such as the movement of object, shake of camera and out-of-focus, the solution spaces of possible functions that map blur image to its sharp counterpart is very large. As a result, image deblurring is a typical ill-posed problem. To address this ill-posed problem, many image deblurring methods have been proposed [1-15].

Traditional shallow image deblurring methods [5-7] apply various constraints (such as uniform [8] or non-uniform [9]) and priors [10] to infer the blur kernel and sharp image, which largely rely on the assumption of blurring model. Besides, since the shallow deblurring methods extract image prior on limited samples, they only achieved good deblurring results on certain blurry images while less robust for real scenarios.

With the success of deep learning over the past decade, some deep convolutional neural networks (CNNs) based approaches [1,11,12] have been introduced to deal with the image deblurring problem. Compared with the shallow methods, deep CNN based image deblurring methods employ a non-linear parametric model to emulate the blurring process of images. Therefore, they can better estimate the blur kernel [13,14]. More recently, the techniques such as RNN [15], LSTM [2] and GAN [11] have also been combined with CNN to accomplish the image deblurring task. Although the aforementioned approaches achieved satisfied image deblurring performance, there still exist some limitations. First, these methods fail to take effective means to reduce the solution space of possible mapping functions from blurry to sharp images. Second, most of them depend on paired blurry and sharp images for training. Nevertheless, paired training data may not always be sufficient in the real-world applications while the unpaired data are easily obtained. Third, the existing image deblurring approaches neglect the constraint on intermedia features of the network, which may deteriorate their deblurring ability.

Recently, dual learning [16-29] has been adopted in computer vision problems. [17,19] proposed DualGAN and CycleGAN respectively, which can be regarded as translators between images in two domains. Cycle consistency is a commonly used criterion in dual learning [19] and has been widely applied to many tasks [20-24]. Although the GAN with cycle consistency can handle the unpaired training samples, the paired samples which provide important information are totally overlooked in them.

In this paper, we introduce a double closed-loop network (DCLNet) to deal with the image deblurring problem. Inspired by the dual learning [17,18,25,26] and cycle consistency [19], our DCLNet constructs a network with double closed-loop structures, which can not only reduce the possible mapping space between sharp and blur image

+Equal contribution authors. *Corresponding authors(wangjz019@nenu.edu.cn). This work was supported by NSFC (61907007), National Key R&D Program of China (2020YFA0714102), Fund of Jilin Provincial Science and Technology Department (20210101187JC).

domains as much as possible, but also better exploit the fine-grained information of images during deblurring.

Our main contributions are three-folds:

1. A double closed-loop network is designed to limit the solution space of mapping from blurry to sharp images and provide constraint on the features obtained by the intermedia layers of network.

2. Without changing network structure, the loss function of our method can be easily extended to deal with the unpaired samples in the training datasets.

3. Both the theoretical analysis and experimental results are provided to demonstrate the effectiveness of our proposed network.

2. THE PROPOSED MODEL

2.1. Architecture design of DCLNet

As shown in Fig. 1, the whole network consists of three parts. Part I is the backbone which serves as a generator to recover a sharp image $\hat{y}$ from the blurry input x. The architecture of backbone contains two main stages. The first stage is a contraction process called encoder which captures the context information of the blurry image by down-sampling. The down-sampling is realized through two convolutional layers with stride as 3 and Relu as active function. The second stage is a symmetric expanding process called decoder which obtains the sharp image by up-sampling the features of encoder. During the up-sampling process of decoder, we need to delicately exploit the fine-grained details to reconstruct the sharp image. Thus, 15 residual channel attention blocks (RCABs) [27] are employed to improve the model capacity, and then the pixel-shuffle [28] is used to upscale the features. Inspired by the U-net [29], we also incorporate skip connections to concatenate the features in encoder and decoder, so that the context information can be well retained.

In part II, an inverse network is designed to map the sharp image y to its blurry counterpart $\hat{x}$. This network can be regarded as a model opposite to the backbone network, thus they form the first closed-loop structure (denoted by green arrow in Fig. 1) to reduce the solution space of our model. The inverse network includes two convolutional layers with stride as 3 and 5 RCABs to estimate the underlying blur kernel and simulate the blurring process. The reason why we adopt the asymmetrical network structure in this closed-loop lies in the fact that learning a map to blur an image is much easier than learning a deblurring map. Here, it should be noted that since the input of this closed-loop in our proposed DCLNet can be either x or y, we can easily extend our approach to handle the unpaired samples in training data.

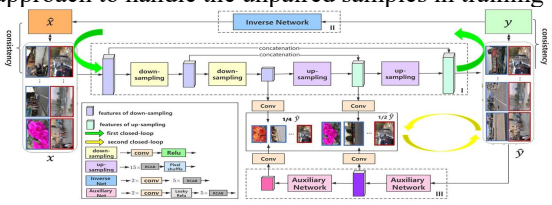

Fig. 1. The structure of the proposed network (DCLNet).

The third part in our model is an auxiliary network contains two convolutional layers with stride as 3, a LeakyRelu and 5 RCABs, which progressively extracts features of the recovered sharp image $\hat{y}$ at different scales. Then, additional convolutional layers are added to the features obtained by auxiliary network and decoder of backbone to produce images at corresponding scales (i.e., 1/2 and 1/4). Through designing a loss function (Equation 5 in Section 2.2) to make the images with the same scale supervise each other, the second closed-loop (denoted by yellow arrow in Fig. 1) is formed to facilitate the recovery of sharp image. Here, we should point out that the auxiliary network is only utilized to constrain the features of decoder part in backbone network. This is because that the encoder in backbone is mainly used to extract context information of the blurry image while the detailed information of sharp image is mostly generated by the decoder.

2.2. The Loss Function

According to our analysis in Section 2.1, the three parts of our network construct three mapping functions between different image domains. Suppose that X is the set of blurry images, Y is the set of sharp images and Y^j is the $1/2^j$ scaled image set obtained by auxiliary network. The three mappings in our model can be denoted as:

$$G=\{X\rightarrow Y\},\ I=\{Y\rightarrow X\},\ Au^j=\{Y\rightarrow Y^j, j=1,2\} \quad (1)$$

where G, I and Au^j are the mapping functions obtained by backbone, inverse and auxiliary networks, respectively.

2.2.1 The Loss function for paired data

Let $\{x_i, y_i\}$ (i =1,…,N) denotes a set of paired training samples in which x_i and y_i are the i-th pair of blurry and sharp images. The loss function of our DCLNet for paired data training can be defined as:

$$\mathcal{L}_{paired}(X,Y)=\mathcal{L}_G+\lambda_1\mathcal{L}_I+\lambda_2\mathcal{L}_{Au} \quad (2)$$

where λ_i (i =1,2) is the tradeoff parameter which can be automatically learned from the data [30]. $\mathcal{L}_G$, $\mathcal{L}_I$ and $\mathcal{L}_{Au}$ represent the loss functions corresponding to the three mappings, respectively. $\mathcal{L}_G$ can be represented as follows:

$$\mathcal{L}_G(X,Y)=\sum_{i=1}^{N}\mathcal{L}_1(G(x_i),y_i)+\mathcal{L}_1(I(G(x_i)),x_i) \quad (3)$$

where $\mathcal{L}_I$ denotes $\mathcal{L}_I$-norm loss. Equation (3) is composed of two terms: the first is used to make the mapping function G generate a sharp image similar to its corresponding ground-truth and the second is a cycle consistency loss to bring x_i back to the original image by the closed-loop structure, so that the solution space can be reduced. $\mathcal{L}_I$ can be represented as follows:

$$\mathcal{L}_I(X,Y)=\sum_{i=1}^{N}\mathcal{L}_1(I(y_i),x_i)+\mathcal{L}_1\big((G(I(y_i)),y_i\big) \quad (4)$$

The first term in Equation (4) is adopted to optimize the mapping function I of the inverse network and the second term is also a cycle consistency loss for solution space restriction. The loss function $\mathcal{L}_{Au}$ of auxiliary network is

$$\mathcal{L}_{Au}(X,Y)=\sum_{i=1}^{N}\sum_{j=1}^{2}\mathcal{L}_1\left(Au^j\big(G(x_i)\big),D_{out\frac{1}{2^j}}(x_i)\right) \quad (5)$$

where $Au^j(G(x_i))$ and $D_{out\frac{1}{2^j}}(x_i)$ represents $1/2^j$ sized images generated by features of the auxiliary network and decoder of backbone network.

2.2.2 The Loss function for unpaired data

In many real-world scenarios, the training dataset for deblurring model may lack of sufficient paired samples. That is, there exist some blurry and sharp images without their counterparts in the training dataset, which will seriously hamper the application of some deblurring methods. Hence, we also extend the loss function in Equation (2) to deal with the dataset contains both paired and unpaired images. The extension of loss function can be defined as:

$$\mathcal{L}_{ext}(X,Y) = k_1\mathcal{L}_{paired} + k_2(\mathcal{L}'_G + \mathcal{L}_{Au}) + k_3(\mathcal{L}'_I + \mathcal{L}'_{Au}) \quad (6)$$

where $\mathcal{L}'_G$ and $\mathcal{L}'_I$ are the modified loss functions of $\mathcal{L}_G$ and $\mathcal{L}_I$ which remove the first term and only retain the cycle consistency loss in Equations (3) and (4). $\mathcal{L}'_{Au}$ is defined as:

$$\mathcal{L}'_{Au} = \sum_{i=1}^{N}\sum_{j=1}^{2}\mathcal{L}_1\left(D_{out\frac{1}{2^j}}(I(y_i)), Au^j(y_i)\right) \quad (7)$$

The parameters k_1, k_2 and k_3 are used to make our model adjust to different situations. Specifically, if the inputs of our model are paired blurry and sharp images, we can degenerate Equation (6) to Equation (2) by setting k_1=1, k_2=0, k_3=0. Contrarily, if a blurry image x is input into our model without its corresponding sharp sample, we can set k_1=0, k_2=1 and k_3=0. Similarly, the parameters are set as k_1=1, k_2=0 and k_3 =1 when only a sharp image y without its blurry counterpart is input.

3. THEORETICAL ANALYSIS

In this section, some theoretical analysis is provided to prove that the closed-loop structure which reduces the possible mapping space between sharp and blurry images can improve the performance of our DCLNet.

We can define three hypotheses for three mappings in our DCLNet as: $G \in \tilde{G}$, $I \in \tilde{I}$ and $Au \in \widetilde{Au}$, where $\tilde{G} = \{G_{\theta_{xy}}(x); \theta_{xy} \in \Theta_{xy}\}$, $\tilde{I} = \{I_{\theta_{yx}}(y); \theta_{yx} \in \Theta_{yx}\}$ and $\widetilde{Au} = \{Au_{\theta_{yy^i}}(y); \theta_{yy^i} \in \Theta_{yy^i},\ i = 1,2\}$, Θ_{xy}, Θ_{yx} and Θ_{yy} are parameter spaces. Through these hypotheses, the function space $\mathcal{H}_{DCL}$ can be defined as $\mathcal{H}_{DCL} \in \tilde{G} \times \tilde{I} \times \widetilde{Au}$. Given the underlying distribution and the loss function in Equation (2), the expected risk to measure the true error of our DCLNet can be defined as:

$$\mathrm{E}(G,I,Au) = \mathrm{E}_{(x,y)\sim\tilde{G}}[\mathcal{L}_G + \lambda_1\mathcal{L}_\mathrm{I} + \lambda_2\mathcal{L}_{Au}], \quad \forall G \in \tilde{G}, I \in \tilde{I}, Au \in \widetilde{Au} \quad (8)$$

However, we can't obtain the true distribution of all samples, so the empirical risk is applied. The empirical risk is the average loss on the training set, which can be defined as:

$$\hat{\mathrm{E}}(G,I,Au) = \frac{1}{N}\sum_{i=1}^{N}\mathcal{L}_G + \lambda_1\mathcal{L}_\mathrm{I} + \lambda_2\mathcal{L}_{Au} \quad (9)$$

Then, the generalization bound is:

$$\mathrm{B}(G,I,Au) = \mathrm{E}(G,I,Au) - \hat{\mathrm{E}}(G,I,Au) \quad (10)$$

From literature [31], we know that the complexity of function space can be measured by Rademacher complexity. Thus, the Rademacher complexity of our DCLNet is defined as follows.

Definition 1. Given the sample set $Z = \{(x_1, y_1), \ldots (x_N, y_N)\}$ drawn from underlying distribution P^N, the Rademacher complexity of DCLNet is:

$$\mathrm{R}_Z(\mathcal{H}_{DCL}) = \mathrm{E}_\mathrm{z}[\widehat{\mathrm{R}_z}(G,I,Au)], \forall G \in \tilde{G}, I \in \tilde{I}, Au \in \widetilde{Au} \quad (11)$$

where $\widehat{\mathrm{R}_z}(G,I,Au)$ is:

$$\widehat{\mathrm{R}_z}(G,I,Au) = E_\sigma[\sup_{(\mathrm{G,I,Au})\in\mathcal{H}_{DCL}}\frac{1}{N}\sum_{i=1}^{\mathrm{N}}\sigma_i\cdot(\mathcal{L}_G + \lambda_1\mathcal{L}_\mathrm{I} + \lambda_2\mathcal{L}_{Au})] \quad (12)$$

where $\sigma = \{\sigma_1, \sigma_2 \ldots \sigma_N\}$ are the random variables with $P\{\sigma_\mathrm{i} = 1\} = P\{\sigma_\mathrm{i} = -1\}$=0.5.

Based on $\mathrm{R}_Z(\mathcal{H}_{DCL})$, we have the following theorem.

Theorem 1. Supposing that the loss function of our network forms a mapping from $X \times Y$ to [0,1]. Then, for any δ>0, with probability at least 1- δ , the following inequation holds for all $(G,I,Au) \in \mathcal{H}_{DCL}$[31].

$$\mathrm{E}(G,I,Au) \le \hat{\mathrm{E}}(G,I,Au) + 2\mathrm{R}_z(\mathcal{H}_{DCL}) + \sqrt{\frac{\log(\frac{1}{\delta})}{2N}} \quad (13)$$

Through Theorem 1, we have:

$$\mathrm{B}(G,I,Au) \le 2\mathrm{R}_z(\mathcal{H}_{DCL}) + \sqrt{\frac{\log(\frac{1}{\delta})}{2N}} \quad (14)$$

In Inequation (14), $\mathrm{R}_z(\mathcal{H}_{DCL})$ and N are the key factors to measure the $\mathrm{B}(G,I,Au)$. That is, the model with more samples and smaller Rademacher complexity has smaller generalization bound. According to the definition of Rademacher complexity, if a network only has a single generator task (i.e., $G \in \tilde{G}$), which reconstruct a sharp image from the blurry input, the Rademacher complexity of it will be larger than our DCLNet. In other words, we have $\mathrm{R}_z(\tilde{G}) > \mathrm{R}_z(\mathcal{H}_{DCL})$ since the capacity of function space $G \in \tilde{G}$ is larger than $\mathcal{H}_{DCL} \in \tilde{G} \times \tilde{I} \times \widetilde{Au}$. Thus, with the same number of samples, the proposed DCLNet enjoys a smaller generalization bound than other approaches without closed-loop structure, which helps our model to give more accurate predictions for the unseen test data.

4. EXPERIMENTS

4.1 Implementation details

We implement our model using PyTorch. Two public datasets (GoPro [1] and HIDE [32]) are employed in this study. Besides, we also provide a real-world dataset called DCLData to compare the performances of various methods. The Adam is used to optimize our network. The learning rate, momentum, momentum2 and weight decay are set as 5e-5, 0.9, 0.999 and 1e-8, respectively. PSNR and SSIM are adopted to evaluate our method. The results of all comparison methods are directly quoted from their corresponding papers.

4.2 Results on GoPro dataset

GoPro dataset consists of 3214 pairs of blurry and sharp images at 720 × 1280 resolution. The training and testing sets include 2103 and 1111 pairs, respectively. We compare the performance of our model with some state-of-the-art

deblurring methods. The quantitative comparisons are shown in Table 1, and some visual comparisons are shown in Fig. 2.

Table 1. Performance comparison on GoPro dataset.

Methods	[1]	[2]	[33]	[4]	[11]	[34]	[35]	[15]	[3]	[36]	[37]	[38]	DCLNet
PSNR	29.08	30.1	30.92	30.21	28.7	29.55	30.28	29.19	32.15	32.02	31.79	31.1	**32.70**
SSIM	0.9135	0.9323	0.9421	0.9345	0.958	0.934	0.912	0.9306	0.956	0.953	0.949	0.9424	0.944

From Table 1, it can be seen that our method outperforms other approaches due to the following reasons. First, the backbone, inverse and auxiliary networks in our DCLNet could guarantee better generalization ability. Second, the auxiliary network imposes constraint on the features of intermedia layers. Thus, more details of the sharp image can be exploited by our DCLNet. Through the visual comparison in Fig. 2, we can also observe that our model consistently produces sharper structures and clearer details than other approaches.

Fig. 2. Visual comparison of the deblurring results on GoPro dataset.

4.3 Results on HIDE dataset

HIDE dataset has 8422 sharp and blurry image pairs. We employ 6397 image pairs for training and 2025 pairs for testing. As shown in Table 2, our method outperforms other methods due to double closed-loop structures. Also, the advantage of our method can be observed from the visual comparison results in Fig. 3.

Table 2. Performance comparison on HIDE dataset.

Methods	[11]	[34]	[2]	[32]	[4]	[36]	DCLNet
PSNR	24.51	26.61	28.36	28.89	29.09	29.98	**30.21**
SSIM	0.871	0.875	0.915	0.93	0.924	0.93	0.931

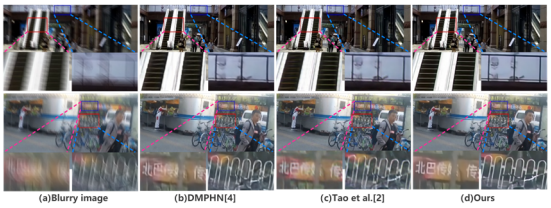

Fig. 3. Visual comparison of the deblurring results on HIDE dataset.

4.4 Results on Real-World dataset with unpaired data.

We also created a real-word dataset called DCLData, which contains 600 paired blurry and sharp images and 1200 unpaired images (600 blurry and 600 sharp) with the size of 720 × 1280. Four different training sample selection schemes (T1-T4) are designed in Table 3 for evaluating the performance of our DCLNet on this dataset. First, thanks to the double closed loop structure in our DCLNet, we can see from Fig. 4 that the proposed model outperforms other methods when only paired data are employed for training. Moreover, from the quantitative result in Table 3 and visual comparison in Fig. 5, we can see that training the proposed model with unpaired data can indeed improve the deblurring performance.

Table 3. Different schemes for training set selection in DCLData set.

Data	T1	T2	T3	T4
Training set	90% paired images	90% paired images + blurry images	90% paired images + sharp images	90% paired images +blurry images+ sharp images
Test set	10% paired images	10% paired images	10% paired images	10% paired images
PSNR	29.11	29.54	29.35	29.87
SSIM	0.901	0.912	0.909	0.916

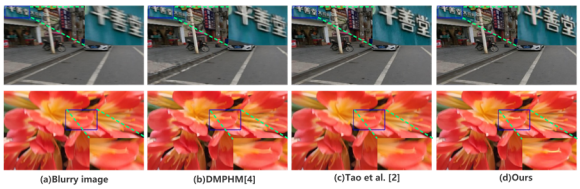

Fig. 4. Visual comparison of the deblurring results obtained by some methods on DCLData dataset under scheme T1.

Fig.5. Visual comparison of different training data selection schemes on DCLData dataset.

4.5 Ablation study and analysis

In this section, several experiments are conducted to validate the effectiveness of individual component in our DCLNet. Through removing some components from our network, we can get three different versions of our model, which are denoted as Net1-Net3 in Table 4. As can be seen from the experimental results in this table, both the inverse and auxiliary networks are essential for our model.

Table 4. Comparison of different versions of our network on GoPro dataset.

Design	Inverse	Auxiliary	PSNR	SSIM
Net1	√	X	31.45	0.928
Net2	X	√	31.53	0.929
Net3	X	X	31.38	0.922
DCLNet	√	√	32.70	0.944

5. CONCLUSION

In this paper, a double closed-loop network was proposed for deblurring task. Different from prior work which merely focused on constructing a single network to straightly recover a sharp image from the blurry one, we introduced two closed-loop structures into our model which could effectively improve the deblurring performance. Moreover, we also extended the loss function of our model to deal with the unpaired samples in the dataset. Through extensive experiments on benchmark datasets and a real-world dataset, the advantage of our network over other state-of-the-art methods has been demonstrated.

6. REFERENCES

[1] Nah, Seungjun, Tae Hyun Kim, and Kyoung Mu Lee. "Deep multi-scale convolutional neural network for dynamic scene deblurring." Proceedings of the IEEE conference on computer vision and pattern recognition. 2017.

[2] Tao, Xin, et al. "Scale-recurrent network for deep image deblurring." Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2018.

[3] Purohit, Kuldeep, and A. N. Rajagopalan. "Region-adaptive dense network for efficient motion deblurring." Proceedings of the AAAI Conference on Artificial Intelligence. Vol. 34. No. 07. 2020.

[4] Zhang, Hongguang, et al. "Deep stacked hierarchical multi-patch network for image deblurring." Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2019.

[5] Shan, Qi, Jiaya Jia, and Aseem Agarwala. "High-quality motion deblurring from a single image." Acm transactions on graphics (tog) 27.3 (2008): 1-10.

[6] Chen, Liang, et al. "Enhanced sparse model for blind deblurring." European Conference on Computer Vision. Springer, Cham, 2020.

[7] Xu, Li, Shicheng Zheng, and Jiaya Jia. "Unnatural l0 sparse representation for natural image deblurring." Proceedings of the IEEE conference on computer vision and pattern recognition. 2013.

[8] Fergus, Rob, et al. "Removing camera shake from a single photograph." ACM SIGGRAPH 2006 Papers. 2006. 787-794.

[9] Zhang, Haichao, and David Wipf. "Non-uniform camera shake removal using a spatially-adaptive sparse penalty." Advances in Neural Information Processing Systems. 2013.

[10] Xu, Zhenhua, Huasong Chen, and Zhenhua Li. "Fast blind deconvolution using a deeper sparse patch-wise maximum gradient prior." Signal Processing: Image Communication 90 (2021): 116050.

[11] Kupyn, Orest, et al. "Deblurgan: Blind motion deblurring using conditional adversarial networks." Proceedings of the IEEE conference on computer vision and pattern recognition. 2018.

[12] Shen, Ziyi, et al. "Deep semantic face deblurring." Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2018.

[13] Sun, Jian, et al. "Learning a convolutional neural network for non-uniform motion blur removal." Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2015.

[14] Pan, Liyuan, et al. "Phase-only image based kernel estimation for single image blind deblurring." Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2019.

[15] Zhang, Jiawei, et al. "Dynamic scene deblurring using spatially variant recurrent neural networks." Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2018.

[16] Liu, Ming-Yu, and Oncel Tuzel. "Coupled generative adversarial networks." Advances in neural information processing systems 29 (2016): 469-477.

[17] Yi, Zili, et al. "Dualgan: Unsupervised dual learning for image-to-image translation." Proceedings of the IEEE international conference on computer vision. 2017.

[18] Guo, Yong, et al. "Closed-loop matters: Dual regression networks for single image super-resolution." Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2020.

[19] Zhu, Jun-Yan, et al. "Unpaired image-to-image translation using cycle-consistent adversarial networks." Proceedings of the IEEE international conference on computer vision. 2017.

[20] Rozumnyi, Denys, et al. "DeFMO: Deblurring and Shape Recovery of Fast Moving Objects." Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2021.

[21] Ma, Jizhou, et al. "Unsupervised multi-class co-segmentation via joint-cut over l1-manifold hyper-graph of discriminative image regions." IEEE Transactions on Image Processing 26.3 (2016): 1216-1230.

[22] Wilson, Kyle, and Noah Snavely. "Network principles for sfm: Disambiguating repeated structures with local context." Proceedings of the IEEE International Conference on Computer Vision. 2013.

[23] Zhou, Tinghui, et al. "Learning dense correspondence via 3d-guided cycle consistency." Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2016.

[24] Godard, Clément, Oisin Mac Aodha, and Gabriel J. Brostow. "Unsupervised monocular depth estimation with left-right consistency." Proceedings of the IEEE conference on computer vision and pattern recognition. 2017.

[25] Wu, Junru, et al. "DAVID: Dual-attentional video deblurring." Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision. 2020.

[26] Xia, Yingce, et al. "Dual supervised learning." International Conference on Machine Learning. PMLR, 2017.

[27] Zhang, Yulun, et al. "Image super-resolution using very deep residual channel attention networks." Proceedings of the European conference on computer vision (ECCV). 2018.

[28] Shi, Wenzhe, et al. "Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network." Proceedings of the IEEE conference on computer vision and pattern recognition. 2016.

[29] Ronneberger, Olaf, Philipp Fischer, and Thomas Brox. "U-net: Convolutional networks for biomedical image segmentation." International Conference on Medical image computing and computer-assisted intervention. Springer, Cham, 2015.

[30] Kendall, A., Y. Gal, and R. Cipolla. "Multi-Task Learning Using Uncertainty to Weigh Losses for Scene Geometry and Semantics." 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2018).

[31] Mohri, Mehryar, Afshin Rostamizadeh, and Ameet Talwalkar. Foundations of machine learning. MIT press, 2018.

[32] Shen, Ziyi, et al. "Human-aware motion deblurring." Proceedings of the IEEE/CVF International Conference on Computer Vision. 2019.

[33] Gao, Hongyun, et al. "Dynamic scene deblurring with parameter selective sharing and nested skip connections." Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2019.

[34] Kupyn, Orest, et al. "Deblurgan-v2: Deblurring (orders-of-magnitude) faster and better." Proceedings of the IEEE/CVF International Conference on Computer Vision. 2019.

[35] Liu, Yiming, et al. "Semantic Information Supplementary Pyramid Network for Dynamic Scene Deblurring." IEEE Access 8 (2020): 188587-188599.

[36] Suin, Maitreya, Kuldeep Purohit, and A. N. Rajagopalan. "Spatially-attentive patch-hierarchical network for adaptive motion deblurring." Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2020.

[37] Jiang, Zhe, et al. "Learning event-based motion deblurring." Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2020.

[38] Zhang, Kaihao, et al. "Deblurring by realistic blurring." Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2020.