Edge Map-guided Scale-iterative Image Deblurring

Sung-Jun Min* and Suk-Ju Kang†
*Sogang University, Seoul, Korea
E-mail: sjmin9868@sogang.ac.kr Tel: +82-10-2388-9868
†Sogang University, Seoul, Korea
E-mail: sjkang@sogang.ac.kr Tel: +82-10-7103-6520

Abstract— Blind image deblurring aims to remove the blur generated from camera motion or moving objects. Advance of deep end-to-end learning methods showed superiority in removing non-uniform motion blur, but there still exists unsharp blurriness due to the pixel-wise loss in a restored image using deep deblurring methods. Therefore, we propose a simple and effective iterative edge map guidance to restore spatial details. The proposed method extracts the edge map of the blurred input image prior to the image deblurring process and restores it to a clear edge map. We use the edge map information for image deblurring task. Therefore, unlike conventional methods, the proposed method performs deblurring by considering the edge information of the clear image, which leads to restored images with sharper edges. Furthermore, the proposed method iteratively reconstructs sharp images and edge maps on multiple scales. This iterative scheme can further improve performance due to the advantages of weight-sharing with multi-scale training. We found that the proposed method showed 0.26dB higher PSNR compared to the original method for GoPro dataset.

I. INTRODUCTION

Blind image deblurring aims to restore a blurred image with little information about the blur kernel. Blur caused by camera shakes, moving objects or low shutter speed not only brings quality degradation of acquired image, but also results in information loss. Therefore, it is essential to remove blurring artifacts and restore image details in vision tasks where clean and sharp images are appreciated.

Prior to the success of deep learning, conventional deblurring methods were studied by applying a variety of constraints to approximate the motion blur kernels, but they require an expensive non-convex nonlinear optimization. Furthermore, designing such priors is a very challenging task, and the commonly used estimation of spatially invariant blur kernel is hard to generalize the real-world motion blur, resulting in a poor deblurring performance of complex blur patterns.

To alleviate this issue, convolutional neural networks (CNNs) were applied to implicitly learn regression between a blurry image input and the corresponding sharp image or to learn general image priors in an end-to-end manner from large-scale data. Among the state-of-the-art architectures for deblurring, the self-recurrent module is widely adopted to restore both low-level contextual feature and high-level spatial details. Nah et al. [1] propose scale-cascaded structure for deblurring. In this coarse-to-fine scheme, finer scale image deblurring is aided by coarser scale features. This coarse-to-fine scheme has been applied in different form of architecture such as multiple-scale network [2, 3] or multiple-patch hierarchical network [5, 6].

However, despite the pros of using the guidance from coarse network to fine network by using this approach, there also exist cons of fixed architecture with longer inference time due to recurrent mechanisms. Also, scaling a blurred image into lower resolution in multi-scale network structure results in loss of information. Furthermore, splitting a blurred input image into multiple patches is not suitable for handling non-uniform blur in dynamic scenes. Therefore, their proposals show some inadequacies when applied to various real-world blurred images. This paper addresses major following challenges of deep deblurring. 1) Following similar pipeline of the coarse-to-fine scheme, multiple-scale or scale-recurrent architecture results in the large number of parameters in training and results in longer inference time. 2) In the process of generating low resolution blurred image, the information loss is inevitable and deblurring performance without any guidance is bound to fail in restoring spatial details like edge feature.

We propose a novel architecture, the gradient-guided self-iterative upscaling network, to overcome the issues, with following key components. First, we utilize gradient features to restore a sharp image from a blurred image input to boost performance of high-level features like edge information. By exchanging features of edge map information extracted from blurred and sharp images to the original deblurring path, our model can restore better edges by compensating the limitation of pixel-wise loss that most multi-scale models suffer. Second, we adopt the iterative scheme across multi scales, which allows us to have fewer trainable parameters than scale-cascaded structure. Compared with previous fixed-level architectures, our network shows flexibility by applying different iterations for training and test by using shared weights. Our main contributions are summarized as follows:

- We utilize the extracted gradient features from blurred image inputs and estimate the corresponding gradient features to restore sharper edges and compensate the blurriness from pixel-wise loss.

- The sharp image and the edge map are predicted iteratively across multi scales by using shared weights, which enables to flexibly apply different iterations for training and test.

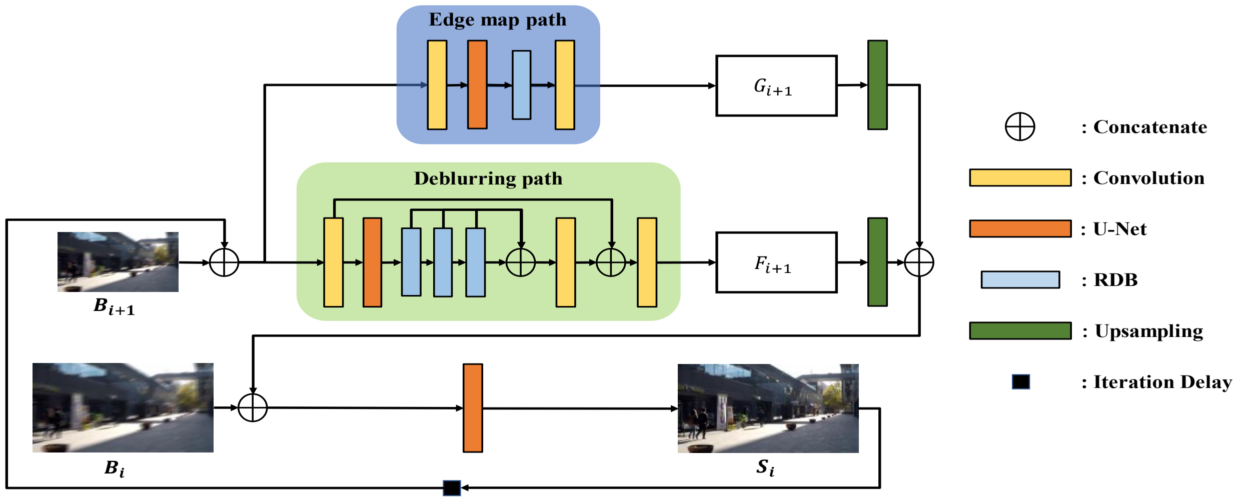

Fig. 1 Overall architecture of scale-iterative upscaling network with edge map guidance (SUNG).

II. Proposed Method

Fig. 1 gives the architecture of our proposed scale-iterative upscaling network with edge map guidance (SUNG). SUNG consists of two modules, feature extraction module and image restoration module. The feature extraction module consists of two paths, a deblurring path and an edge path. For an I-iteration deblurring process, we first generate a set of pyramid blurred images B_i $(i = I, I-1, \cdots, 0)$ with B_i denoting a $1/2^i$ down-sampled image with corresponding restored output image S_i. The deblurring process starts from the smallest scale, $i = I$. At the very beginning, where we do not have the predicted sharp image, we assume $S_I = B_I$ for the 0^{th} output.

Feature extraction module consists of a deblurring path and an edge map path. The deblurring path performs a typical deblurring operation with input pair of blurred images, B_{i+1}, and corresponding predicted sharp image, S_{i+1}. At the same time, edge map of same blurred and predicted sharp image pair is fed to the edge map path. The features extracted from each path are F_{i+1} and G_{i+1}. These two output features are upscaled and concatenated with the next blurred input image B_i. By applying this concatenated feature to the U-Net structure, restored sharp image S_i is obtained. Then S_i is concatenated with the next blurred input B_i, which becomes the next blurred and sharp image input pair for the iterative process.

By iteratively repeating the above process, the size of the restored sharp image is upscaled until it reaches full resolution. This full resolution image is taken as the final output.

A. Deblurring Path

We used the deblurring path first proposed by Ye et al. [6]. The deblurring path is constructed by using a modified residual dense network (RDN) [7], combined with a U-Net structure with skip connections between features across the overall deblurring path. A pair of blurred and sharp images is concatenated as an input of the deblurring path. After a convolution layer and U-Net structure, a feature with 32 channels is passed through the 3 residual dense blocks (RDBs) in order. These features are then concatenated and passed through a convolution layer, which is again concatenated with the first output feature of the convolution layer with skip connection. The output feature of deblurring path F_{i+1} has is then upscaled for the next process.

B. Edge Path

The edge path is to estimate the translation of edge maps from the blurred modality to the sharp one. By applying a convolution layer with a fixed kernel, an edge map of an image is easily obtained by computing the difference of the adjacent pixels. By focusing on the intensity, not the direction, of the edge map, we can obtain sharpness in an image. Therefore, an edge map can be regarded as another kind of image, and the image-to-image translation technique can be applied. We utilized this edge map information to introduce an auxiliary loss, which will be explained in the section D.

The edge map path generates the output feature G_{i+1} by the input of edge map from blurred and sharp image pair, unlike the deblurring path where a pair of the blurred and sharp images is fed. It passes through a convolution layer, U-Net, and a single RDB, and another convolution layer. Single RDB was enough to extract edge information of blurred image to restore edge map of corresponding sharp image. With guidance of G_i, we could predict images with sharper edges. Visual comparison results are shown in Fig. 2.

C. Image restoration module

Three features are concatenated to estimate the sharp image S_i in the image restoration module. G_i is the upscaled feature output of the edge path, F_i is the upscaled feature output of the deblurring path, and B_i is the next blurred image. The concatenated feature is then fed to another U-Net to estimate

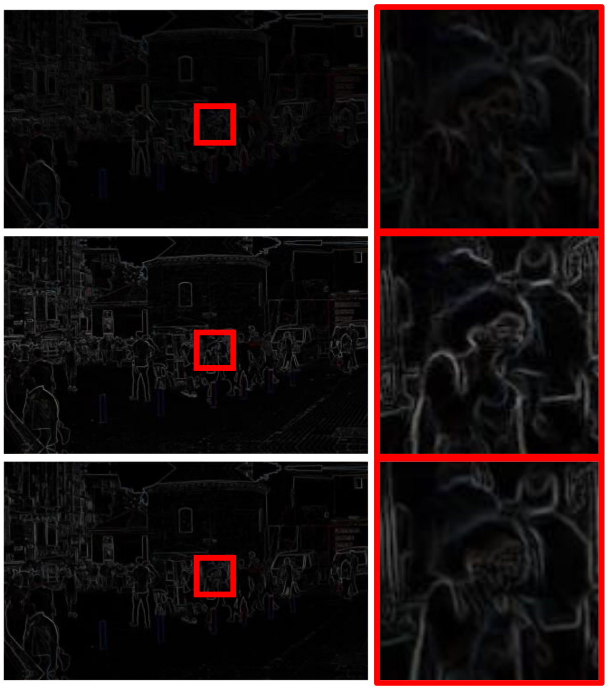

Fig. 2 Visual comparison of edge map. From top to bottom, each row depicts edge map extracted from blurred image, ground truth image, and estimated sharp image by SUNG, S_i.

the sharp image. The output feature S_i is the restored image of the i^{th} iteration, and it is concatenated with B_{i-1} to extract features in feature extraction module for the next iteration.

D. Objective Functions

Conventional Loss: Most deblurring methods optimize their networks by a common pixelwise loss, which is efficient for task of blind image deblurring by PSNR. To minimize the pixel difference between the ground-truth image and the blurred input image and to accelerate convergence of training, the pixel loss is widely used:

$$L^{Pix} = || S_i - GT_i ||_1, \tag{1}$$

where GT denotes the ground-truth image. Unlike these benefits of using pixel loss, overall visual coherency may decrease since it is based on PSNR. Thus, the result may not be able to restore sharp edges that needs to be restored. Therefore, using the pixel loss alone is not suitable for deblurring task, and we introduce the following edge loss to alleviate this issue.

Edge Loss: First, we adopt an auxiliary edge loss between edge maps of predicted sharp image $M(S_i)$ and edge maps of ground-truth sharp image $M(GT_i)$, to utilize edge information in the image restoration module as a guidance to restore sharper edges:

$$L^{EdgeAux} = || M(S_i) - M(GT_i) ||_1. \tag{2}$$

Table. 1 Comparisons on GoPro dataset with state-of-the-art methods. The best performance is shown in red and second-best is in blue.

Method	GoPro (PSNR/SSIM)
Tao et al. [3]	30.25 / 0.9030
Zhang et al. [5]	30.45 / 0.9057
Ye et al. [6]	30.21 / 0.9041
Ours (SUNG)	30.47 / 0.9047

Note that an output feature of edge map path is used to generate an input feature for image restoration module. To better reconstruct predicted sharp image S_i with sharper edge, supervision of edge map feature G_i is essential, and it is learned to mimic the edge maps of ground-truth image, $M(GT_i)$. Therefore, we apply supervision edge loss for successful feature extraction of edge map path:

$$L^{EdgeSup} = || G_i - M(GT_i) ||_1. \tag{3}$$

Overall Objective: L^{Pix} is used to optimize in a pixel-wise level, as most conventional deblurring task does. $L^{EdgeAux}$ is used to restore the blurred image to mimic its edge to sharp image edges. This is done only with the guidance of $L^{EdgeSup}$, which is to generate high-quality edge maps with given blurred image edge map. The overall objective is defined as follows:

$$L = L^{Pix} + L^{EdgeAux} + L^{EdgeSup}. \tag{4}$$

III. EXPERIMENTS

A. Implementation Details

Training, Validation, and Test Datasets: We trained and evaluated our methods on GoPro dataset [1], which contains large-scale real-world blurred and sharp image pairs. GoPro dataset consists of 3214 pairs of blurred and sharp images with 720 × 1280 resolution. 2103 image pairs were used for training and remaining 1111 pairs were used for testing.

Training and Test Details: For training, we trained the network with iteration = 3. We used Adam optimizer with $\beta_1 = 0.9$, $\beta_2 = 0.999$, and $\varepsilon = 10^{-8}$. We also used 256 × 256 patch randomly cropped from GoPro training dataset as the input with a batch-size of 16. 500 epochs with the learning rate decay rate of 0.5 was applied every 100 epochs. The model was implemented on PyTorch framework and trained on GEFORCE RTX 3090 GPUs. Benefiting from shared weights of iterative scheme, we adopted iteration = 4 in test phase for our best performance.

B. Experimental Results

We compared our method with other state-of-the-art methods on the GoPro dataset. We chose Tao et al. [3] and Zhang et al. [5] for comparisons. Table. 1 shows the results of comparison with these state-of-the-art methods on benchmark dataset. For evaluating Tao et al. [3], we used their "color"

Fig. 3 Visual comparison on benchmark datasets. From top to bottom are blurry input, deblurring results of Tao et al. [3], Zhang et al. [5], Ye et al. [6], and ours

model among (lstm/gray/color) models released by the author. For evaluating Zhang et al. [5], we used their best 1-2-4-8 DMPHN model. The PSNR and SSIM evaluation on GoPro

Table. 2 Comparison with same model size w/ and w/o edge map

	PSNR	SSIM
Base	30.17	0.9038
Our proposed model	30.47	0.9047

Fig. 4 Comparisons between proposed model with base model. From left to right, GT images, ours, and base model images.

dataset are calculated by MATLAB built-in function. For evaluating baseline model Ye et al. [6], best learning strategy with 3 iterations was adopted.

From Table. 1, our SUNG model shows better performance compared to other methods on GoPro dataset. Our model was slightly better than Zhang et al. [5] in PSNR and had the comparable performance in SSIM. Also, compared with similar structure using iterative scheme, our model performed better in both PSNR and SSIM. Although they have the similar structure, our model performed better by using the guidance of edge map information. Visual comparison results on benchmark dataset, GoPro dataset, are shown in Fig. 3.

IV. Ablation Study

We further studied effectiveness of our edge map guidance with the edge loss. To show that the performance enhancement was due to our edge map guidance, not the deepness of the model, we constructed a model with same depth as our model but without the edge map guidance. We call this base model, by adding two convolution layers with U-Net and RDBs to the deblurring path to make the same model size of our edge map guidance model SUNG. The qualitative and quantitative results between base model and our model are shown in Fig. 4 and Table. 2.

Fig. 4 shows the comparison of the base model and our model with edge guidance. These images cropped from

GoPro test dataset for qualitative evaluation. Compared to the results of the base model, our results restored the finer details in the textures and restored edge or the letters more clearly. Furthermore, as shown in Table. 2, model with edge map guidance showed 0.3dB higher performance even with the same model size. Therefore, it is shown that the performance improvement comes from the edge map guidance structure rather than the depth of a network.

V. Conclusion

In this paper, we propose a scale-iterative upscaling network with edge map guidance (SUNG). Our model extracts gradient features from blurred image inputs and estimates the corresponding gradient features to restore sharper edges to compensate the blurriness from pixel-wise loss. Also, our model predicts the sharp images and the edge maps iteratively in multi scales by using shared weights. This allows our model to achieve flexibility by applying different iterations for training and test. As a result, our model achieves 0.26dB better performance than the base model without using the edge map feature, and this scheme can be applied to other models or tasks to aim for performance enhancement.

Acknowledgement

This work was supported by LG Electronics and by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (No. 2021R1A2C1004208 and No. 2020M3H4A1A02084899).

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] 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.

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

[4] 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.

[5] 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.

[6] Ye, Minyuan, Dong Lyu, and Gengsheng Chen. "Scale-iterative upscaling network for image deblurring." *IEEE Access* 8 (2020): 18316-18325.

[7] Zhang, Yulun, et al. "Residual dense network for image restoration." *IEEE Transactions on Pattern Analysis and Machine Intelligence* 43.7 (2020): 2480-2495.