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

EGNet: Enhanced Gradient Network for Image Deblurring

Changdi Zhao · Xiaoguang Di · Feng Gao

Received: date / Accepted: date

Abstract In recent years, convolutional neural networks have been increasingly developed in the field of image deblurring. However, there are still many problems in the existing methods. First, most methods lack attention to the middle layers of the networks, resulting in the lack of gradient information which is important but cannot be restored well by the existing methods. And the ability of the networks which can reconstruct sharp images is reduced. Second, the existing deblurring methods can not perform the self-ensemble of the networks, which has been proved to further improve the network performance in many fields such as image classification and single image super-resolution. Third, the loss function of the existing networks all directly use MAE loss, MSE loss or adversarial loss, which lack the special design for edge information and structural information in the images. To solve the above problems, we enhanced the gradient information of the feature maps after the Encoder in the network, then designed the Self-Ensemble network to further improve the performance, and finally we redesigned the loss function to adaptively redistribute the weight of each position in the gradient map according to the input image. Through the extensive experiments, our proposed method achieve the state-of-the-art results of image deblurring on the existing dataset, and also outperform other methods in generalization ability on RealBlur dataset as well as real blurry images.

Keywords Image Deblurring · Enhanced Gradient Network · Self-Ensemble Network

Changdi Zhao
E-mail: 18846144928@163.com

Xiaoguang Di(corresponding author)
E-mail: dixiaoguang@hit.edu.cn

Feng Gao
E-mail: 21s104157@stu.hit.edu.cn

1 Introduction

The objective of image deblurring is to remove the blurry artifacts. Given the input blurry image, it aims to estimate a sharp image which contains more sharp details. And image deblurring can greatly improve the accuracy of other computer vision tasks such as image classification, object detection and semantic segmentation, etc.

The commonly-used blur model is described as Eqn.(1).

$$v_i = k_i \otimes u_{nn(i)} + n_i. \tag{1}$$

Where v is the blurry image, u is the sharp image of size $H \times W$, k is a set of per-pixel blur kernels k_i of size $K \times K$, $\otimes$ represents the convolution operation, $u_{nn(i)}$ represents a window of size $K \times K$ around pixel i in image u and n_i is additive noise[**1**].

The traditional non-blind deblurring methods[**2**] restore the sharp image from the existing blurry image on the premise that the blur kernel k is known. So it cannot be used in practice. In recent years, convolutional neural networks have been further developed, showing superior performance in various computer vision tasks. Nah et al.[**3**], Tao et al.[**4**] and Gao et al.[**5**], Zhang et al.[**6**], Zamir et al.[**7**], Mao et al.[**8**] and Cho et al.[**9**] performed the end-to-end image deblurring through a multi-stage and multi-patch neural networks, achieving the state-of-the-art results.

For the existing image deblurring methods based on deep neural networks, there are still many problems that have not been solved. First of all, in each stage, only the Skip-Connection between the Encoder and the Decoder is performed, but the gradient information in the middle feature map which is encoded by the Encoder is missing. Therefore, the deblurring performance of the network is limited. Secondly, the existing methods do not perform the self-ensemble such as averaging multiple results and Geometric Self-ensemble, etc. As a result, the network cannot aggregate the advantages of multiple results simultaneously. Thirdly, MAE Loss or MSE Loss is often used as the content

loss or gradient loss. In the calculation process, the pixel-level weight allocation is missing, so it decreases the subjective image quality.

According to the above problems, we have made a series of improvements. Our main contributions are summarized as follows:

(1) We designed an enhanced gradient network to increase the proportion of gradient information in the feature map. Through the network, we can achieve the purpose of adaptively improving the correlation between the feature map and the sharp image. And the enhanced gradient network can be embedded into the existing methods such as MPRNet[**7**], DMPHN[**6**] et al., to further generate more accurate results spatially.

(2) We designed a self-ensemble network to fuse multiple test results, such as the multi-patch test results, the multi-scale test results, etc. And our self-ensemble network can outperform State-Of-The-Art (SOTA) networks in image restoration performance.

(3) We proposed a new loss function, which redistributes the weight of gradient information extracted from different pixels in the image, and the proposed loss function can adaptively improves the proportion of the gradient information in the image. And we can get contextually-enriched results.

(4) Experiments are carried out to verify the generalization ability of the network on the public dataset and the real blurry images captured by our mobile phones. The effectiveness of the enhanced gradient network is verified through the Hilbert Schmidt Independence Criterion(HSIC).[**10**].

2 Related Work

During the imaging process, due to the out-of-focus of the imaging device and the relative motion between the objects with the imaging devices, the captured images will contain the blurs, such as the out-of-focus blur, motion blur et al. Existing SOTA learning-based methods[**6**][**7**][**8**][**9**] directly perform the end-to-end learning through the convolutional neural networks, greatly improving the image restoration performance. Nowadays, the learning-based image deblurring methods can be classified into three categories,i. e. CNN-based methods, GAN-based methods and self-ensemble methods.

CNN-Based methods. Carbajal et al.[**1**] proposed a general nonparametric model for dense non-uniform motion blur estimation. Zhang et al.[**6**] did not use the multi-scale input at different stages, but used multi-patch input at different stages to improve the deblurring effect for small objects. Zamir et al.[**7**] continued the idea of multi-scale network and multi-patch network, and introduced an attention mechanism in the fusion process of each stage. Quan et al.[**11**] proposed a single image defocus deblurring network by unrolling a fixed-point iteration derived from a GKM-based model of defocus blurring process. Quan et al.[**12**] proposed a non-blind image deblurring approach which is built upon the unfolded optimization of a deconvolution model equipped with a Gabor-domain denoising prior. Chen et al.[**13**] proposed a dataset-free approach which is not dependent on external training dataset. Quan et al.[**14**] developed a scheme of constructing non-local wavelet frame and tight frame which are adaptive to the input image. However, these methods do not consider the gradient information in the feature maps of the intermediate layers, so the deblurring results contain over-smooth contents. Therefore, the network performance is limited.

GAN-Based methods. On the basis of WGAN[**15**][**16**], Kupyn et al.[**17**] only used adversarial loss and perceptual loss to train their network DeblurGAN. Since it does not directly use MAE loss or MSE loss, it has a poor performance on PSNR. To solve this problem, Kupyn et al.[**18**] made further improvements and proposed the DeblurGAN-v2. They combined the rich semantic information among the feature maps with different resolution. And they proposed a new loss function RaGAN-LS loss. It combines the existing MSE loss and perceptual loss, so the restored image with both high PSNR and high SSIM is obtained. But in general, the images generated by GAN-based methods often have some artifacts.

Self-Ensemble methods. The self-ensemble method can not be applied well in the existing deblurring methods. But in the image classification task, the aggregation of multiple results through the multi-scale and multi-crop testing, random horizontal flipping[**19**][**20**] can effectively improve the accuracy of the model. When it is used to the image deblurring, the restored image will contain overly smooth contents and artifacts. In single-image super-resolution, Geometric Self Ensemble[**21**] uses a single trained model to obtain 8 augmented inputs by flipping and rotation. And the average are performed to obtain the final result, which can effectively improve the performance. But it is only valid for symmetric downsampling methods. So it can not be applied to image deblurring directly. Chen et al.[**22**] proposed an ensemble approach to exploit the priors from untrained neural network for blind image deblurring, which aggregates the deblurring results of multiple untrained neural network for improvement. In [**23**], the prediction is done by averaging over the estimates from both perturbed the images and kernels. These methods use kernels to maintain the consistency of noise characteristics between the training and test process, whitch limits the generalization ability and robustness of the deblurring network.

There are some problems in the existing deblurring methods. Most of the methods use Encoder-Decoder network directly and do not consider the gradient information in the intermediate feature maps, so the deblurring results contain overly smooth contents. And the deblurring network can not directly use the existing self-ensemble methods to boost the network performance. Our proposed network can effectively increase the proportion of gradient information in the intermediate feature map and the proposed

self-ensemble network can boost the network deblurring performance.

3 Proposed Method

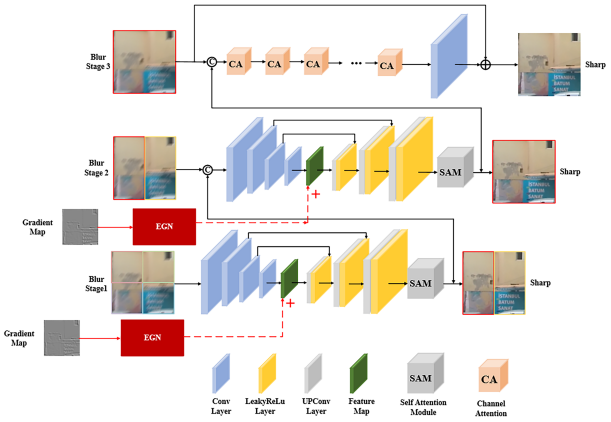

Fig. 1: The overall architecture of our proposed method.

The overall framework of our image deblurring network consists of three stages to progressively restore the images, as shown in Figure 1. The first two stages are based on encoder-decoder subnetworks that learn the broad contextual information due to large receptive fields. And the last stage employs a subnetwork without any downsampling operation. The input and output of the three stages are blurry images and restored images, respectively. We adapt the multi-patch hierarchy on the input image and split the image into non-overlapping patches: four for stage-1, two for stage-2, and the original image for the last stage, as shown in Fig.1.

In addition, instead of simply using Encoder-Decoder subnetworks, we incorporate an Enhanced Gradient Network between the encoder and decoder, that increases the proportion of gradient information in the feature map, which will be detailed in Subsection 3.1. On this basis, we adopt a variety of test methods, mainly from the following aspects: 1) input the blurry image which is downsampled – image deblurring – output the restored image and the up-sampling; 2) input the blurry image – image deblurring – output the restored image; 3) input the blurry image which is splited into four non-overlapping patches – image deblurring – output the restored patches. And these test methods are used in Subsection 3.2. As for the loss function, we propose a new gradient loss in Subsection 3.3. Next, we describe each key element of our method.

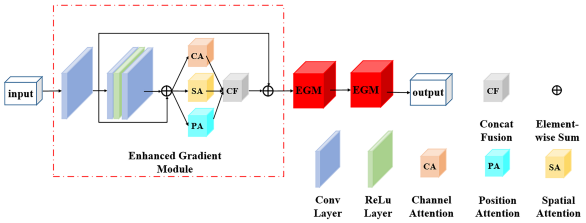

Fig. 2: Enhanced Gradient Network.

3.1 Enhanced Gradient Network

The Enhanced Gradient Network(EGN) is shown in Figure 2. In the stage-1 and stage-2 of the network, in addition to taking the blurry image as input, the gradient map of the image is also put into the Enhanced Gradient Network. By doing this, we can improve the proportion of gradient information in the feature map which is obtained by the Encoder.

3.1.1 Enhanced Gradient Module

EGN includes several Enhanced Gradient Modules(EGMs). The structure of EGM is shown in Figure 2. Here, we use residual channel attention module and residual spatial attention module[**24**] to learn the channel and pixel weights of the features. Further, we exploit the position attention module to enhance the local feature correlation among the feature maps. We experimentally demonstrate that these attention modules boost the performance as detailed in Sec. 4.

3.1.2 Residual Channel Attention

The Residual Channel Attention(RCA) module is shown in Figure 3. The input of RCA module goes through successive convolutional layers to get $x^{in_1} \in R^{H\times W\times C}$, as shown in Eqn.(2). Where in_1 represents the input to the module of weight distribution, $H \times W$ denotes the spatial dimension and C represents the number of channels.

$$x^{in_1} = C_2(R(C_1(x^{in}))). \tag{2}$$

Where C_1 and C_2 represent the convolutional layers in RCA module. R represents the activation layer.

Next, we redistribute the weight of each channel of x^{in_1}. As shown in Eqn.(3), after we get the feature map of the residual branch, we add it with the input x^{in} to get the final result x^{out}.

$$x^{out} = x^{in} + x^{in_1} * C_4(R(C_{c3}(GA(x^{in_1})))). \tag{3}$$

Where GA represents Global Average Pooling. C_{c3} and C_4 represent the third and the fourth convolutional layers in RCA module, respectively. $*$ represents element-wise product.

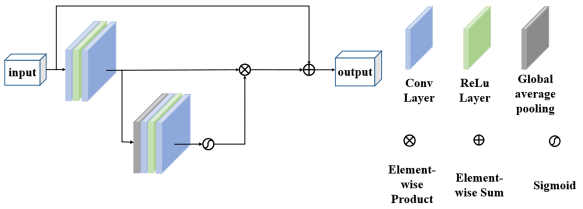

Fig. 3: Residual Channel Attention Module.

3.1.3 Residual Spatial Attention

In Residual Spatial Attention(RSA) module, as shown in Figure 4, we can get x^{in_1} like the RCA module. Then we redistribute the pixel weights and increase the pixel weights that are conducive to image restoration. As shown in Eqn.(4), x^{in_1} is sent to an average pooling layer and a max pooling layer to get $A \in R^{H\times W\times 1}$ and $M \in R^{H\times W\times 1}$. We can get the weight coefficient for each pixel by sending the concatenation of A and M to a convolutional layer and a sigmoid activation function.

$$x^{out} = x^{in} + x^{in_1} * \sigma(C_{s3}(Cat(AP(x^{in_1}), MP(x^{in_1})))). \quad (4)$$

Where C_{s3} represents the third convolutional layer in RSA module. $Cat(a,b)$ represents the concatenation of a and b. AP and MP represent the average pooling layer and the max pooling layer respectively. σ represents the sigmoid activation function.

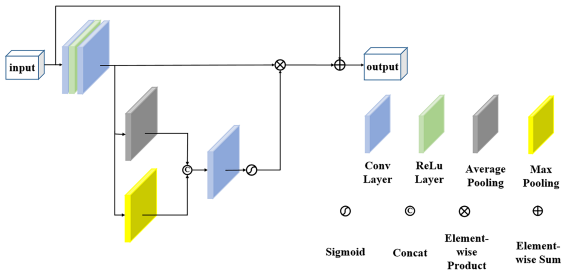

Fig. 4: Residual Spatial Attention Module.

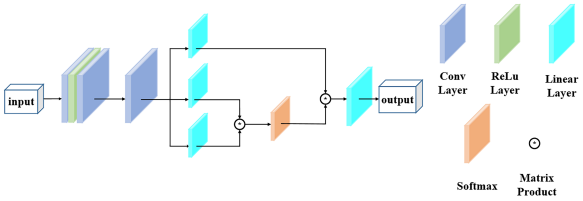

Fig. 5: Position Attention Module.

3.1.4 Position Attention

In the Position Attention(PA) Module, as shown in Figure 5, $x^{in_1} \in R^{H\times W\times C}$ is reshaped to $R^{HW\times C}$. And we can get $q \in R^{HW\times C_1}$, $k \in R^{C_1\times HW}$ and $v \in R^{HW\times C_2}$ by three linear layers. Then we can get coefficient matrix $z \in R^{HW\times HW}$ by the matrix multiplication between q and k. The coefficient matrix z and v are multiplied to get the output $v \in R^{HW\times C_2}$. Finally, we get the output $x^{out} \in R^{H\times W\times C}$ by a linear layer and the reshaping operation(as shown in Eqn.(5)-(9)).

$$q = L_1(C_{p3}(x^{in_1})). \quad (5)$$

$$k = L_2(C_{p3}(x^{in_1})). \quad (6)$$

$$v = L_3(C_{p3}(x^{in_1})). \quad (7)$$

$$z = SM(MUL(q,k)). \quad (8)$$

$$x^{out} = L_4(MUL(z,v)). \quad (9)$$

Where C_{p3} represents the third convolutional layer in PA module. L_1, L_2, L_3, L_4 represent the four linear layers in PA module. MUL represents the matrix multiplication. $SM(a)$ represents the softmax result of a.

3.2 Self-Ensemble Network

After obtaining a variety of test results, such as the output obtained by downsampling, the output obtained by the original image, etc, we find that simply averaging multiple test results cannot effectively combine edges of the test results and may have checkerboard artifacts. Therefore, it is necessary to design a Self-Ensemble network. We can obtain the final result through a Self-Ensemble Network, which aims to combine the advantages of various test results.

As shown in Figure 6, the Self-Ensemble network is a multi-branch channel attention network. The multiple input images of the network are I_1, I_2, $I_3 \ldots I_n$. The image I_i is passed through a residual block to obtain the feature map I_{i_1}, which is described as Eqn.(10).

$$I_{i_1} = I_i + C_2(R(C_1(I_i))). \quad (10)$$

The feature maps of each branch are added and passed through several convolutional layers and Softmax layers(as shown in Eqn.(11)). $z_{in} \in R^{H\times W\times n}$ is used as the input of Softmax layer, and the weight coefficients of different branches are obtained after the softmax operation(as shown in Eqn.(12)).

$$z_{in} = C_5(R(C_4(R(C_3(\sum_{i=1}^{n} I_{i_1}))))). \quad (11)$$

$$z_{out}^{i,j,c} = \frac{exp(z_{in}^{i,j,c})}{\sum_{m=1}^{n} exp(z_{in}^{i,j,m})}. \quad (12)$$

Where $z_{in}^{i,j,c}$ represents the element whose coordinate is (i,j)

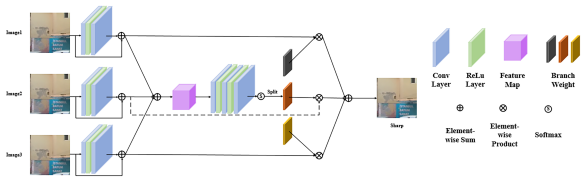

Fig. 6: Self-Ensemble Network.

in the c-th channel of the z_{in}. n represents the number of images.

We split z_{out} to get the corresponding coefficient matrix $z^i, i \in (1,n)$ on each branch. Finally, the coefficient matrix z^i is dot-multiplied with its corresponding input image I_i to get the output results of the network.

3.3 Loss Function

The loss function of our network mainly includes three parts, i.e. content loss, gradient loss and SSIM loss. And we will describe these parts in detail.

3.3.1 Content Loss

We use content loss to ensure minimizing the pixel-to-pixel differences between the restored image with the ground truth. For content loss, we use the L2 loss as shown in Eqn.(13).

$$L_{cont}(I,\hat{I}) = \frac{1}{H \times W} \sum_{i=0}^{W-1} \sum_{j=0}^{H-1} (I(i,j) - \hat{I}(i,j))^2. \tag{13}$$

Where I represents the sharp image. $\hat{I}$ represents the restored image.

3.3.2 Gradient Loss

In order to further improve the proportion of gradient information in the restoration results, we also use gradient loss in the loss functions. It ensures that the gradient of the restored image is consistent with the ground truth. As shown in Eqn.(14), we use L2 loss as the gradient loss.

$$L_{grad}(I,\hat{I}) = \frac{1}{H \times W} \sum_{i=0}^{W-1} \sum_{j=0}^{H-1} (\nabla I(i,j) - \nabla \hat{I}(i,j))^2. \tag{14}$$

Where ∇ represents the Laplacian operator, the proposed gradient extraction operator is designed as Eqn.(15).

$$\nabla I = G(F(Ave(I))) * G(F(I)). \tag{15}$$

Where I represents the image. $Ave(\cdot)$ represents the mean filter operation. $F(\cdot)$ represents the Laplacian operator. $G(\cdot)$ stands for normalization operation.

Using Eqn.(15) for gradient extraction has two advantages. First, it can suppress the influence of noise. Second, it can improve the weight of the gradient information adaptively. As shown in Figure 7, it is proved that the quality of the gradient map can be effectively improved. And many gradients that are invisible previously can be effectively extracted.

(a) (b) (c)

Fig. 7: The comparison results of gradient maps which are extracted by different gradient extraction operators. (a) Input image. (b) The result extracted by the Laplacian operator. (c) The gradient map extracted by Eqn.(15).

3.3.3 SSIM Loss

To enable the network to learn more contextually-enriched results, we use the MS-SSIM loss. For the restoration results R and the ground-truth G, we calculate the mean u_R, u_G, the standard deviation σ_R, σ_G and the covariance σ_{RG}. Then we can calculate MS-SSIM loss according to Eqn.(16).

$$L_{ssim} = 1 - \prod_{m=1}^{M} \left(\frac{2u_R u_G + c_1}{u_R^2 + u_G^2 + c_1}\right)^{\beta_m} \left(\frac{2\sigma_{RG} + c_2}{\sigma_R^2 + \sigma_G^2 + c_2}\right)^{\gamma_m}. \tag{16}$$

Where m represents the different scales. c_1 and c_2 can maintain the stability during training. β_m and γ_m represent the relative importance.

3.3.4 Overall Loss

We use a combination of content loss, gradient loss, and SSIM loss to construct the overall loss function of the network, as shown in Eqn.(17).

$$Loss = \sum_{i=1}^{S} (\alpha L_{cont} + \beta L_{grad} + \gamma L_{ssim}). \tag{17}$$

Where the hyperparameter is α=1, β=0.05, γ=0.1.

4 Experiment

In this section, we first introduce the dataset which is used to verify the effectiveness of our method. Secondly, we describe our experimental configuration. Thirdly, we compare the performance of our method with the existing SOTA methods in dataset and real-world scenarios. Fourthly, we conduct the ablation experiments to verify the effectiveness of the proposed models.

4.1 Datasets

4.1.1 GOPRO

Nah et al.[**3**] collected high-frame rate video and then performed motion-blur generation through the multi-frame synthesis. The entire dataset contains 3214 pairs of blur-sharp images, of which 1111 pairs of images are used for test set and 2103 pairs of images are used for training set.

4.1.2 RealBlur

Unlike the GOPRO dataset, the RealBlur[**25**] dataset is collected in the real world and mainly contains two subsets: RealBlur-J is formed with the JPEG images, and RealBlur-R is generated by the RAW images. The dataset contains a total of 4738 pairs of images, of which 980 pairs of images are used for test set and 3758 pairs of images are used for training set.

4.2 Experiment Settings

The training process of the network is divided into two stages. In the first stage, we do not train EGN and train the network according to the training strategy in [**7**], the patch size of the input is 256×256, the batch size is set to 1, and the network is trained for 6×10^6 iterations. For data augmentation, we randomly flip the input horizontally or vertically. We train the network using the Adam optimizer with an initial learning rate of 2×10^{-4}, which is gradually reduced to 1×10^{-6}. In the second training stage, we fix the network parameters which are obtained from the first training stage, and train the EGN separately. The hyperparameters are the same as the first stage.

4.3 Comparison with the state-of-the-arts(SOTA) method

We compare our proposed method with the existing SOTA methods, which include DeblurGAN-v2[**18**], MPRNet[**7**], MIMONet[**9**], etc. We embed EGN into the SOTA method MPRNet. In order to verify the generalization ability of the network, we apply the model to train on the GOPRO dataset and test on the RealBlur dataset. The test results on the GOPRO dataset and the RealBlur dataset are shown in TABLE 1. 'R-J' and 'R-R' represent 'RealBlur-J' and 'RealBlur-R', respectively. The best results are shown in bold, and the second best results are shown in the underlined form. It can be seen from TABLE 1 that our method achieves the best results in terms of PSNR and SSIM, which further prove that our network has excellent generalization ability. The visualization results are shown in Figure 8 to Figure 11. Also, we compare the run-time complexity and model complexity with other models, which are tested on the GOPRO dataset. The test results on the GOPRO dataset are shown in TABLE 2. Our run time is longer than MPRNet, because we embed our model into the MPRNet.

Table 1: The performance comparison of different methods on the GOPRO dataset and the RealBlur dataset.(The model is trained on the GOPRO training set.)

Methods	PSNR(GoPro/R-J/R-R)	SSIM(GoPro/R-J/R-R)
SRN	30.26/26.33/33.62	0.934/0.856/0.946
DMPHN	30.45/25.72/33.62	0.935/0.844/0.943
DeblurGAN-v2	29.55/<u>26.68</u>/33.41	0.934/0.862/0.936
BANet	32.44/-/-	0.957/-/-
MIMO-UNet+	32.44/26.01/33.65	0.957/0.851/0.945
[**26**]	31.23/-/-	0.946/-/-
MPRNet	<u>32.66</u>/26.51/<u>33.91</u>	<u>0.959</u>/<u>0.865</u>/<u>0.949</u>
Ours	**33.00/26.96/34.30**	**0.961/0.877/0.952**

Table 2: The run-time complexity comparison of different methods on the GOPRO dataset.(The model is trained on the GOPRO training set.)

Methods	Run Time(s)	Model Size(MB)
DMPHN	0.135	29.01
MPRNet	0.217	20.13
MIMO-UNet+	0.028	16.11
DeblurGAN-v2	0.115	28.27
[**26**]	0.790	26.34
Ours	0.231	26.60

Finally, we use mobile phone to capture the images in real indoor and outdoor scenes. We also compare the processing results with the other methods. Since the captured images do not have the corresponding sharp images, we only show qualitative comparisons. As shown in Figure 12, our images are much sharper and contain much more gradient information.

4.4 Ablation Study

In this section, we further analyze the effectiveness of our proposed network through ablation study, mainly from the following aspects: 1) the input of EGN; 2) the attention mechanism in EGN; 3) effectiveness of EGN; 4) effectiveness of Gradient Loss.

4.4.1 The input of EGN

Inspired by YOLOR[**27**], we design various inputs for EGN, i. e. random initialization with a mean value of 0 or 1, which is added or multiplied to the network. It is tested on the GOPRO dataset, and the obtained results are shown in TABLE 3. In TABLE 3, the 'add' and 'mul' mean to fuse with the

Table 3: Comparative experimental results on the GOPRO dataset when using different inputs for EGN.

Input	PSNR(w A-w/o A)	SSIM(w A-w/o A)
add+random(0,0.02)	32.9088/32.9081	0.9609/0.9608
add+random(0,0.2)	32.9088/32.9080	0.9609/0.9609
mul+random(1,0.02)	32.9080/32.9068	0.9608/0.9608
Ours	**32.9250/32.9154**	**0.9610/0.9609**

Table 4: The comparative experimental results of the network with and without EGN on the GOPRO dataset.

	PSNR	SSIM
w/o EGN	32.52	0.958
w EGN	**32.92**	**0.961**

Table 5: The HSIC comparison of the results on the GOPRO dataset with and without EGN.

	blur	sharp
w/o EGN	1	1.6838
w EGN	1	**1.8134**

feature map in the form of addition and multiplication, respectively. Random(a,b) means a normal distribution with mean 'a' and standard deviation 'b'.

It can be seen from TABLE 3 that our network has good adaptability to different forms of input, and can get good results on the GOPRO dataset. In TABLE 3, 'w A' and 'w/o A' represent 'with Attention' and 'without Attention', respectively. It is proved that our network is very robust to the different inputs. And EGN can adaptively learn more weights which are beneficial to image deblurring.

4.4.2 The Attention Module in EGN

In order to verify the effectiveness of the attention mechanism in EGN. We redesign EGN and make it only contain continuous convolution layers. The experimental results tested on the GOPRO dataset are shown in TABLE 3.

It can be seen from TABLE 3 that a variety of network structures can get good results, but the best results are obtained by using the attention module. The effectiveness of the attention module is confirmed.

4.4.3 The effectiveness of EGN

We first compared the experimental results of the network with and without EGN. Secondly, HSIC[**10**] is used to evaluate the correlation between the images and the feature maps

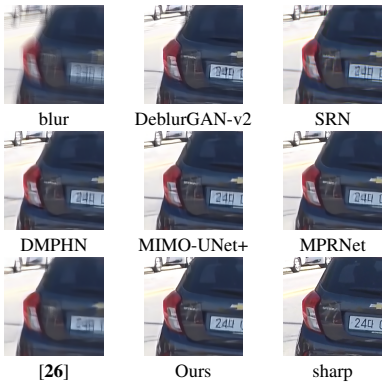

Fig. 8: The comparison of the results between the SOTA methods with our proposed method on the GOPRO dataset.

Fig. 9: The comparison of the results between the SOTA methods with our proposed method on the GOPRO dataset.

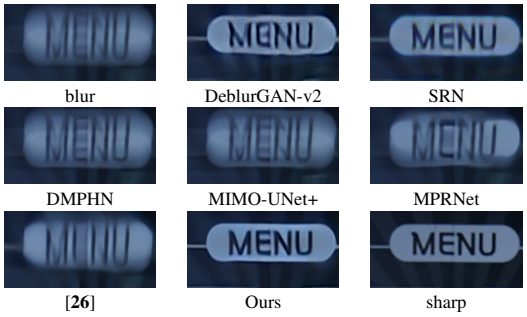

Fig. 10: The comparison of the results between the SOTA methods with our proposed method on the RealBlur dataset.

in the middle layer of the network, which proves the effectiveness of EGN.

For the input feature maps of the last convolutional layer, we use HSIC as an independent metric to evaluate the correlation between the feature maps and the images. In TABLE 5, the correlation between the feature maps and the blurry image is set to 1, so the HSIC value of the sharp image is greater than 1, indicating that correlation between the feature maps and the sharp image is higher.

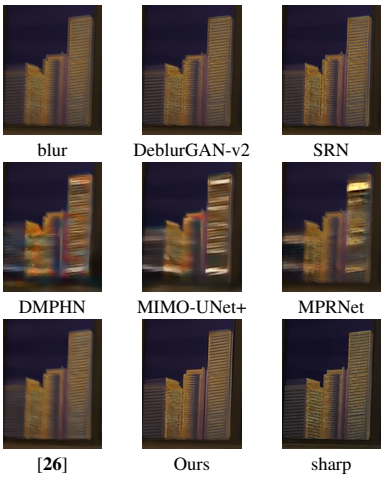

Fig. 11: The comparison of the results between the SOTA methods with our proposed method on the RealBlur dataset.

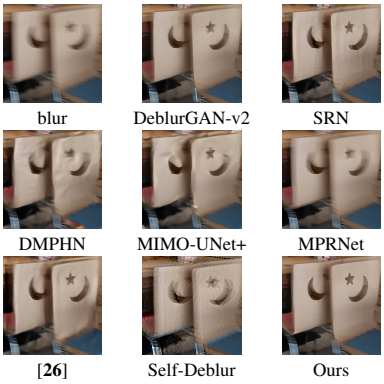

Fig. 12: The comparison of the results between the SOTA methods with our proposed method on real-shot blurry images.

As can be seen from TABLE 4 and TABLE 5, after the introduction of EGN, the network performance has been significantly improved and the similarity between the feature maps and the sharp image will be significantly improved, which is the main reason why the introduction of EGN is effective for image deblurring.

4.4.4 The effectiveness of Gradient Loss

We performed ablation study of the proposed gradient loss, which was trained on the GOPRO dataset and then tested on the RealBlur dataset. The quantitative results are shown in TABLE 6. The representative visualization results are shown in Figure 13 and Figure 14.

As can be seen from TABLE 6, Figure 13 and Figure 14, after the introduction of Gradient Loss, the network has better generalization ability.

Table 6: The comparison of the results with and without Gradient Loss.

	PSNR(R-R/R-J)	SSIM(R-R/R-J)
w/o Gradient Loss	33.7142/26.3708	0.9464/0.8617
w Gradient Loss	34.2960/26.9638	0.9520/0.8765

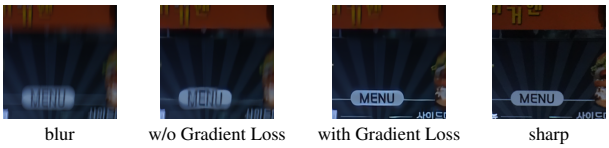

Fig. 13: The results of the ablation study of the Gradient Loss on the RealBlur dataset.

Fig. 14: The results of the ablation study of the Gradient Loss on real-shot blurry images.

5 Conclusion

In this paper, a novel image deblurring method based on Enhanced Gradient Network(EGN) is presented. The proposed EGN can increase the proportion of gradient information in the feature maps and improve the correlation between the feature maps and the sharp image. It can also eliminate the artifacts and retain the sharp details. In order to fuse multiple test results and recover more edge details, we further propose a self-ensemble network and redesign the loss functions. It can get better ensemble results than the other self-ensemble methods. The experiments implemented on public datasets and real blurry images demonstrate that our proposed method achieves the state-of-the-art results in both qualitative and quantitative ways.

Acknowledgements

This work was supported by the Natural Science Foundation of Heilongjiang Province, China, under Grant LH2021F026 and Fundamental Research Funds for the Central Universities under Grant HIT.NSRIF202243.

References

1. Carbajal, Guillermo, et al. "Single image non-uniform blur kernel estimation via adaptive basis decomposition." Computing Research Repository (CoRR), arXiv: 2102.01026, pp. 1-11, feb 2021 (2021).
2. Boden, A. F., et al. "Massively parallel spatially variant maximum-likelihood restoration of Hubble Space Telescope imagery." JOSA A 13.7 (1996): 1537-1545.

3. 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.
4. Tao, Xin, et al. "Scale-recurrent network for deep image deblurring." Proceedings of the IEEE conference on computer vision and pattern recognition. 2018.
5. 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.
6. 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.
7. Zamir, Syed Waqas, et al. "Multi-stage progressive image restoration." Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2021.
8. Mao, Xintian, et al. "Deep residual fourier transformation for single image deblurring." arXiv preprint arXiv:2111.11745 (2021).
9. Cho, Sung-Jin, et al. "Rethinking coarse-to-fine approach in single image deblurring." Proceedings of the IEEE/CVF international conference on computer vision. 2021.
10. Ma, Wan-Duo Kurt, J. P. Lewis, and W. Bastiaan Kleijn. "The HSIC bottleneck: Deep learning without back-propagation." Proceedings of the AAAI Conference on Artificial Intelligence. Vol. 34. No. 04. 2020.
11. Quan, Yuhui, Zicong Wu, and Hui Ji. "Gaussian Kernel Mixture Network for Single Image Defocus Deblurring." Advances in Neural Information Processing Systems 34 (2021): 20812-20824.
12. Quan, Yuhui, et al. "Nonblind image deblurring via deep learning in complex field." IEEE Transactions on Neural Networks and Learning Systems (2021).
13. Chen, Mingqin, et al. "Nonblind Image Deconvolution via Leveraging Model Uncertainty in An Untrained Deep Neural Network." International Journal of Computer Vision (2022): 1-20.
14. Quan, Yuhui, Hui Ji, and Zuowei Shen. "Data-driven multi-scale non-local wavelet frame construction and image recovery." Journal of Scientific Computing 63.2 (2015): 307-329.
15. Goodfellow, Ian, et al. "Generative adversarial networks." Communications of the ACM 63.11 (2020): 139-144.
16. Gulrajani, Ishaan, et al. "Improved training of wasserstein gans." Advances in neural information processing systems 30 (2017).
17. Kupyn, Orest, et al. "Deblurgan: Blind motion deblurring using conditional adversarial networks." Proceedings of the IEEE conference on computer vision and pattern recognition. 2018.
18. Kupyn, Orest, et al. "Deblurgan-v2: Deblurring (orders-of-magnitude) faster and better." Proceedings of the IEEE/CVF International Conference on Computer Vision. 2019.
19. Hu, Jie, Li Shen, and Gang Sun. "Squeeze-and-excitation networks." Proceedings of the IEEE conference on computer vision and pattern recognition. 2018.
20. Xie, Saining, et al. "Aggregated residual transformations for deep neural networks." Proceedings of the IEEE conference on computer vision and pattern recognition. 2017.
21. Lim, Bee, et al. "Enhanced deep residual networks for single image super-resolution." Proceedings of the IEEE conference on computer vision and pattern recognition workshops. 2017.
22. Chen, Mingqin, et al. "Self-Supervised Blind Image Deconvolution via Deep Generative Ensemble Learning." IEEE Transactions on Circuits and Systems for Video Technology (2022).
23. Quan, Yuhui, et al. "Learning Deep Non-blind Image Deconvolution Without Ground Truths." European Conference on Computer Vision. Springer, Cham, 2022.
24. Woo, Sanghyun, et al. "Cbam: Convolutional block attention module." Proceedings of the European conference on computer vision (ECCV). 2018.
25. Rim, Jaesung, et al. "Real-world blur dataset for learning and benchmarking deblurring algorithms." European Conference on Computer Vision. Springer, Cham, 2020.
26. Xu, Yong, et al. "Attentive deep network for blind motion deblurring on dynamic scenes." Computer Vision and Image Understanding 205 (2021): 103169.
27. Wang, Chien-Yao, I-Hau Yeh, and Hong-Yuan Mark Liao. "You only learn one representation: Unified network for multiple tasks." arXiv preprint arXiv:2105.04206 (2021).

Declarations

- **Ethics approval and consent to participate**

  Not applicable

- **Consent for publication**

  Not applicable

- **Availability of data and materials**

  Not applicable

- **Competing interests**

  The authors declare that they have no competing interests.

- **Funding**

  This work was supported in part by the Natural Science Foundation of Heilongjiang Province of China (No.LH2021F026) and Fundamental Research Funds for the Central Universities(No. HIT.NSRIF202243).

- **Authors' contributions**

  Changdi Zhao and Xiaoguang Di designed the research. Changdi Zhao drafted the manuscript. Xiaoguang Di helped organize the manuscript. Changdi Zhao, Xiaoguang Di and Feng Gao revised and finalized the paper.

- **Acknowledgements**

  We appreciated the Natural Science Foundation of Heilongjiang Province of China (No.LH2021F026) and Fundamental Research Funds for the Central Universities(No. HIT.NSRIF202243) for their support.

- **Authors' information (optional)**

  Changdi Zhao received the B.S. degree in automation from Harbin Institute of Technology, China, in 2020. He is currently a graduate student at the Control and Simulation Center, Harbin Institute of Technology. His current research interests include deep learning, object detection and image restoration and enhancement.

  Xiaoguang Di (corresponding author) is currently a Professor with the Control and Simulation Center, Harbin Institute of Technology. His current research interests include image restoration and enhancement, object detection and recognition, deep learning, and SLAM. He is a member of the Chinese Association of Automation and the China Simulation Federation.

  Feng Gao received the B.S. degree in automation from Harbin Engineering University, China, in 2021. He is currently a graduate student at the Control and Simulation Center, Harbin Institute of Technology. His current research interests include deep learning, object detection and image restoration and enhancement.