IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Under revision for resubmission.

Digital Object Identifier –.—–/ACCESS.2019.DOI

Multi-Scale Neural Network with Dilated Convolutions for Image Deblurring

JOSE JAENA MARI OPLE[1], PIN-YI YEH[1], SHIH-WEI SUN[3], I-TE TSAI[1], and KAI-LUNG HUA[1,2].
[1]Department of Computer Science and Information Technology, National Taiwan University of Science and Technology, Taipei 10607, Taiwan
[2]Center for Cyber-Physical System Innovation, National Taiwan University of Science and Technology, Taipei 10607, Taiwan
[3]Department of New Media Art, Taipei National University of the Arts, Taipei 112, Taiwan

Corresponding author: Kai-Lung Hua (hua@mail.ntust.edu.tw)

ABSTRACT In image deblurring, information from the regions where the blur was propagated is needed for effective deblurring. For example, large blurs, such as those caused by fast-moving objects leaving a trail of afterimages, need spatial context from a large region, while small blurs, such as those caused by slight camera shake, need spatial context from a smaller scope. In this paper, we used multi-scale features to provide the spatial dependencies needed to deblur non-uniform blurs. Compared to previous works, we efficiently extract multi-scale features using two approaches: (1) coarse-to-fine scheme that can extract multi-scale features by applying the network to different scales of the images, and (2) dilated convolutions that can extract multi-scale features by using different dilation rates. Combining both methods has a multiplicative effect since multi-scale features from dilated convolutions are extracted from the input images at different scales (i.e. coarse-to-fine scheme). Furthermore, we optimized our network architecture by using parallel convolutions to decrease the execution time of the deblurring process. We show that our proposed method has better results than state-of-the-art methods in terms of image quality and execution time.

INDEX TERMS Blind motion deblurring, convolutional neural network, dilated convolution, multi-scale information, coarse-to-fine network.

I. INTRODUCTION

A blurry image has missing or distorted information that could be useful for personal viewing or other image processing tasks. Blind motion deblurring is the process of removing blur from images without knowledge of the blurring process. Conventionally, the blurring process is defined as the convolution of a sharp image and a blur kernel to produce the blurry image so that the blur kernel is approximated rather than the entire sharp image [1]. However, this definition assumes that the blur kernel is uniform for the entire image, and it fails if the blurs are non-uniform (i.e. different from region to region). Uniform blurs could happen because of camera shake hence the entire image is affected. On the other hand, non-uniform blurs are common in dynamic scenes where moving objects cause motion blurs. If object motion information is available, it is easier to remove motion blurs; however, this is not always the case. Some methods try to handle non-uniform blurs by estimating motion flow [2] or by performing region segmentation then deblurring each region using a different blur kernel [3], [4].

Rather than finding the blur kernel then estimating the sharp image, end-to-end deblurring networks learn the mapping from the input blurry image to the latent sharp image, and these methods [5]–[7] have shown success in single image deblurring, even if the images have non-uniform blur. The architecture of these deep learning approaches usually contains deep stacks of convolutional layers due to the following reasons. First, a deep network can represent a more complex mapping from the input blurry image to the latent sharp image compared to a shallow network. And second, the convolutional operation is a local operation and nested convolutional operations are needed so that the features can have a global receptive field. However, deep stacks of convolution layers increase the number of parameters and the execution time. Furthermore, when deblurring images with intense non-uniform blurs, ringing artifacts could sometimes be found.

Moving objects produce blur that could be characterized by the intensity of their movement. For example, fast-moving objects produce a large blur indicated by a long trail of afterimages towards the direction of its motion. On the other hand, subtle camera shake or slow object movement can produce blurs localized in a small region. Images with non-

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.2980996, IEEE Access

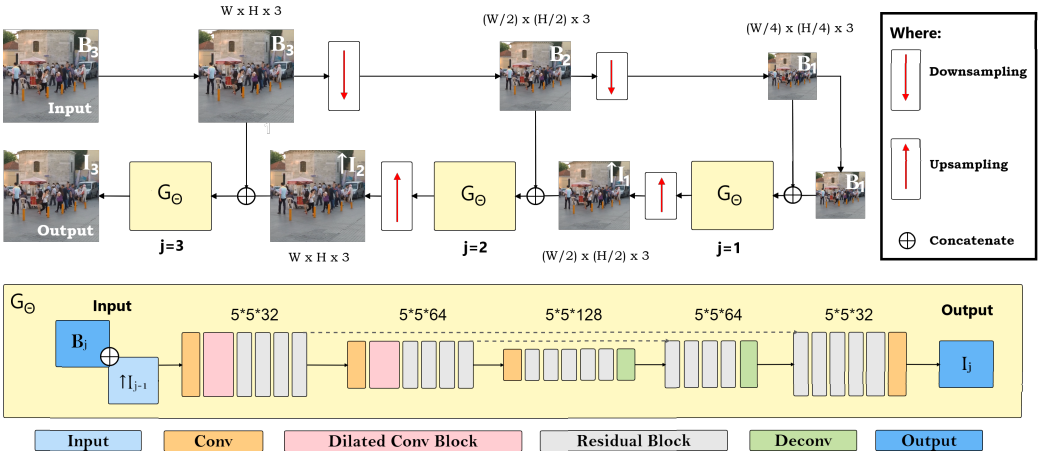


FIGURE 1. Our network G_θ is applied to different scales of the blurry image input. B_j, I_j, j denote input blurry image, output sharp image, and j-th scale level, respectively. The resolution at scale j=3 is the same as the input resolution.

uniform blurs contain different sized blur caused by motions of varying intensity, and different sized blurs require different spatial dependencies to remove. Small blurs require spatial information from nearby positions while intense blurs, that usually cause the ringing artifacts, require spatial information from a larger scope. We decided to design our network to handle these using multi-scale features.

The focus of our paper is to create an end-to-end image deblurring network that can deblur images with non-uniform blurs using multi-scale features, and with faster execution time using network optimization. We only used the blurry and sharp image pairs to train our deep learning network, and no further auxiliary information (e.g. object motion) was used.

To represent the complex mapping from blurry image to sharp image using fewer parameters, we decided to adopt the architecture of Scale Recurrent Network. [7], which uses a U-Net architecture [8] and the coarse-to-fine scheme [5], [7], [9].

The U-Net architecture [8] is an encoder-decoder architecture with residual connections between an encoder and their corresponding decoder. The encoder-decoder architecture allows deeper networks with fewer parameters because of the downsampling layers in the encoder blocks. The coarse-to-fine scheme iteratively processes the input image at different scale levels, from low (coarse) to high (fine) resolution. This strategy is effective for both deep learning and traditional methods [5], [7], [9], and processing the image in different scales can extract multi-scale features. However, spatial information is lost in the encoder block because of the downsampling and adding more scale levels in the coarse-to-fine scheme is inefficient if we want more multi-scale features.

Our solution is to use parallel dilated convolution [10], [11], with different dilation rates, in the encoder block. The advantages of this solution is manyfold: (1) reduces the spatial information lost during downsampling in the encoder blocks, (2) allows global receptive field at fewer layers, (3) extracts multi-scale feature because of the different dilation rates, (4) has a multiplicative effect with the coarse-to-fine scheme (e.g. using multi-scale features at different images in different scales), (5) allows parallel optimization to reduce the execution time.

The main contributions of this work are summarized as follows:

- We implemented a network that could extract multi-scale features efficiently, then use these features to effectively deblur images, even with non-uniform blur.
- We optimized the network architecture by using parallel convolutions that result in faster execution time of the deblurring process.
- We show that our proposed method performs favorably against state-of-the-art deblurring methods in terms of image quality and execution time.

II. RELATED WORKS

Traditional methods for image deblurring define the blurry image as the convolution of a sharp image and a blur kernel so that the blur kernel is approximated rather than the entire sharp image [1]. However, this is an ill-posed assumption since both the blur kernel and the sharp image are unknown; thus infinite solutions exist. Traditional methods use constraints, and prior knowledge of the blur kernel and input image to reduce the number of solutions, but most of these approaches require manual parameter-tuning and expensive computation [12]–[16]. Furthermore, blurs could vary from pixel to pixel, and segmentation is usually performed to estimate non-uniform blurs [3], [4].

In recent years, there is a significant success for deep learning in image processing, particularly in convolutional neural networks (CNN). CNN was shown to have excellent performance in blind motion deblurring [2], [5]–[7], [17]–[20]. Sun et al. [18] used CNN to estimate the blur kernel similar to traditional methods. Some approaches use CNN to learn effective priors for image deblurring. Gong et al. [2] used a fully convolutional network to estimate the motion flows of scene objects. Other CNN approaches, restore the sharp image by finding the mapping from the input blurry

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.2980996, IEEE Access

image to the target sharp image. Mao et al. [17] used a feedforward network with multiple layers of residual blocks. Li et al [20] used coarse-to-fine deblurring of the input image and discriminative priors that classify blurry image regions. Nah et al. [5] proposed Multi-Scale CNN (MSCNN) to restore latent sharp images progressively from coarse-to-fine scales of the blurry input image. Tao et al. [7] proposed Scale-Recurrent Network (SRN) proposed a new structure that combines U-net [8] and Recurrent Neural Network (RNN) to restore blurry images from low resolution to full resolution. Zhang et al. [6] proposed Spatially Variant RNN (SVRNN) that uses an encoder-decoder architecture and RNN to deblur images. Kupyn et al. [19], used Generative Adversarial Network (GAN) and perceptual loss [21] to deblur images.

III. PROPOSED METHOD

Our primary goal is to generate a sharp image from the input blurry image using multi-scale features. We used aggregated dilated convolutions along with coarse-to-fine scheme to extract the features.

A. MULTI-SCALE FEATURES

Images with non-uniform blurs, such as dynamic scenes filled with moving objects, contain different regions with varying blur intensity and direction. Blurs with varying intensity require spatial information to areas where the blur was propagated. For example, small blurs need spatial information from the nearby positions while intense blurs need information from a large region. We decided to use multi-scale features to remove the blurs of different intensities. Our method of generating multi-scale features is done using coarse-to-fine scheme and aggregated dilated convolution. Both methods can extract multi-scale features by themselves; however, combining the two methods have a multiplicative effect by extracting multi-scale features (i.e. aggregated dilated convolutions) at different scales of the images (i.e. coarse-to-fine scheme).

1) Coarse-to-Fine Scheme

Coarse-to-fine scheme is an approach where a method is progressively applied to the input image at different scales, from low (coarse) to high (fine) resolution. It is an effective approach, which is used by traditional and deep learning methods [5], [9], [18], for extracting multi-scale features. It could also save parameter count since it could emulate a deep network by applying the network multiple times, and the earlier iterations are cheaper to compute since the resolution is smaller.

In this paper, we defined the number of scales M as 3, and the maximum image resolution $H_M \times W_M$ as 720×1280. The scales are labelled as j, where $j \in \{1, 2, ..., M\}$. Each scale j is half the resolution of the succeeding scale. For $M = 3$ and $H_M \times W_M = 256 \times 256$, the specific resolutions of each scale will be $\{180 \times 320, 360 \times 640, 720 \times 1280\}$. For images that does not fit the aspect ratio, they will be padded using reflection padding.

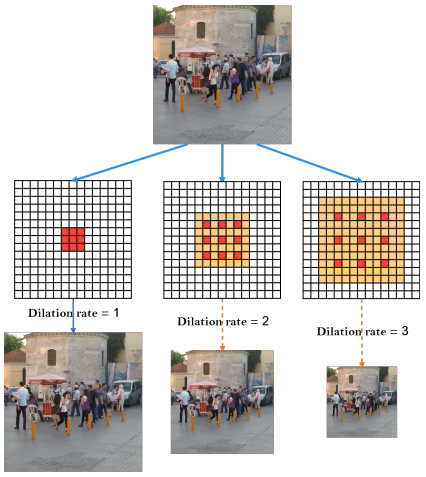


FIGURE 2. Illustration of using different dilation rates to get multi-scale features. Increasing the dilation rate will also increase the receptive field and the output feature map will be smaller if no padding is provided. With $dilation\ rate = 1$, it is the same as standard convolution.

2) Aggregated Dilated Convolutions

Dilated convolutions, also called atrous convolution in some references, are similar to normal convolutions, but instead of using neighboring cells as input to the operation, the input cells are some distance apart. For clarification, Yu et al. [10] called it "dilated convolution," but no dilated filters are represented or constructed.

As seen in Fig 2, using different dilation rates produce feature maps for different scales. Aggregating the output of convolutions with varying dilation rates is easy since the output features have the same sizes, provided that the appropriate padding is used.

The filter size of dilated convolutions can be formulated as:

$$F = (k - 1) * (r - 1) + k \tag{1}$$

where F, k, r are the new filter size, original kernel size, and dilation rate, respectively [10]. With $r = 1$, it is the same as the standard convolution. Increasing the dilation rate will also increase the receptive field. This fact implies that convolution layers could receive features with global receptive field earlier without needing deep stacks of convolutional layers.

More information about the dilated convolution block used in the network can be found in Section III-B3.

B. NETWORK ARCHITECTURE

Our network G_θ is based on SRN [7], which is an encoder-decoder architecture with residual blocks. We modified the original network by adding our dilated convolution block and using parallel optimizations.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.2980996, IEEE Access

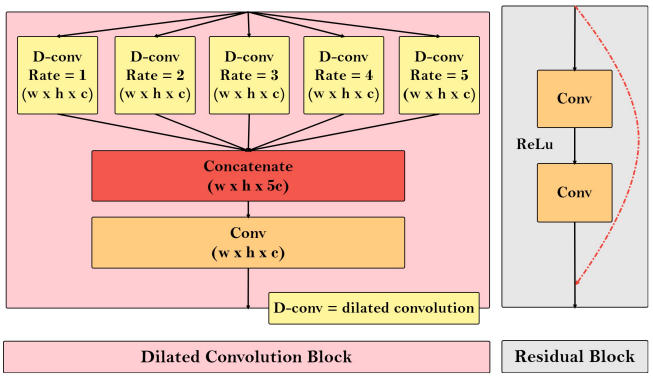


FIGURE 3. (a) **Dilated Convolution Block.** The dilated convolution block contains parallel convolutional layers with different dilation rates (1 to 5), to handle different receptive fields. (b) **Residual Block.** The residual block we used has a simpler architecture compared to ResNet [22]. The batch normalization is removed [5], [23] and there is no activation function before the final output [24].

1) Input

Since our method uses coarse-to-fine scheme, our network G_θ produces intermediate outputs I_j for scale j. The network G_θ will receive two inputs: the downscaled blurry image B_j, and the upscaled intermediate output of the previous scale I_{j-1}. The two inputs will be concatenated, and the result will be the actual input to network G_θ. For the coarsest scale ($j = 1$), where there is no previous intermediate output, we define $I_0 = B_1$.

2) Encoder-Decoder Structure

An encoder-decoder network [8], [17] is a CNN architecture that consists of two components: encoder and decoder. The encoder converts the input data into feature maps that have smaller spatial sizes but have more channels, while the decoder transforms its input to feature maps with larger spatial but with fewer channels. For example, the encoder can shuffle a feature map with size $w \times h \times c$ to a feature map with size $\frac{w}{2} \times \frac{h}{2} \times 2c$ in the encoder while the decoder can shuffle a feature map from size $\frac{w}{2} \times \frac{h}{2} \times 2c$ back to $w \times h \times c$. This kind of structure allows networks with more layers since the decreasing spatial sizes reduce the number of parameters, and additional layers can represent a more complex mapping. Furthermore, the bottleneck in the middle of the architecture motivates the network to learn the important features.

The overall structure of our encoder-decoder network could be seen in Fig 1. The encoder blocks consist of a convolution layer, a dilated convolution block, and residual blocks. In the encoder blocks, we use convolution with $stride = 2$ to downsample the feature maps, and we also double the channel size. In the decoder block, residual blocks were used and transpose convolutions were used to upsample the feature maps. There are residual connections to the decoder block from their corresponding encoder block.

3) Dilated Convolution Block

Other than the advantages stated in Section III-A2, dilated convolution has good synergy with the encoder-decoder architecture. In the encoder block, downsampling reduces the feature size and it could lose some spatial information. However, multi-scale features could still contain coarse-level spatial information in some of its channels; thereby, mitigating the information loss after the feature size reduction. Furthermore, The parallel convolutions in the dilated convolution block decrease the execution time of the deblurring process during test time since the dilated convolution layers are independent and can be processed simultaneously.

The architecture of our dilated convolution block could be found in Fig 3 (a). We concatenated the output of five convolutions with different dilation rates. Then we used the aggregated output of the parallel layers to produce a feature map with similar size as the input.

4) Residual Learning

Residual connections make it easier to train deeper networks by speeding up convergence and avoiding gradient vanishing during backpropagation. Furthermore, the blurry input image and the ground-truth sharp image share many similarities, such as color and overall structure, and makes the network learn the differences between the two images rather than build a sharp image from the ground up.

Our residual block architecture could be found in Fig 3 (b). Different from the original paper [25], our residual block is designed to be optimized for image generation and has fewer parameters. We remove the batch normalization since these layers add more parameters and are not suitable for image generation tasks since it limits the range flexibility of features and produces unwanted artifacts [5], [23]. We also removed the activation function before the last layer of the convolution block according to He et al. [24].

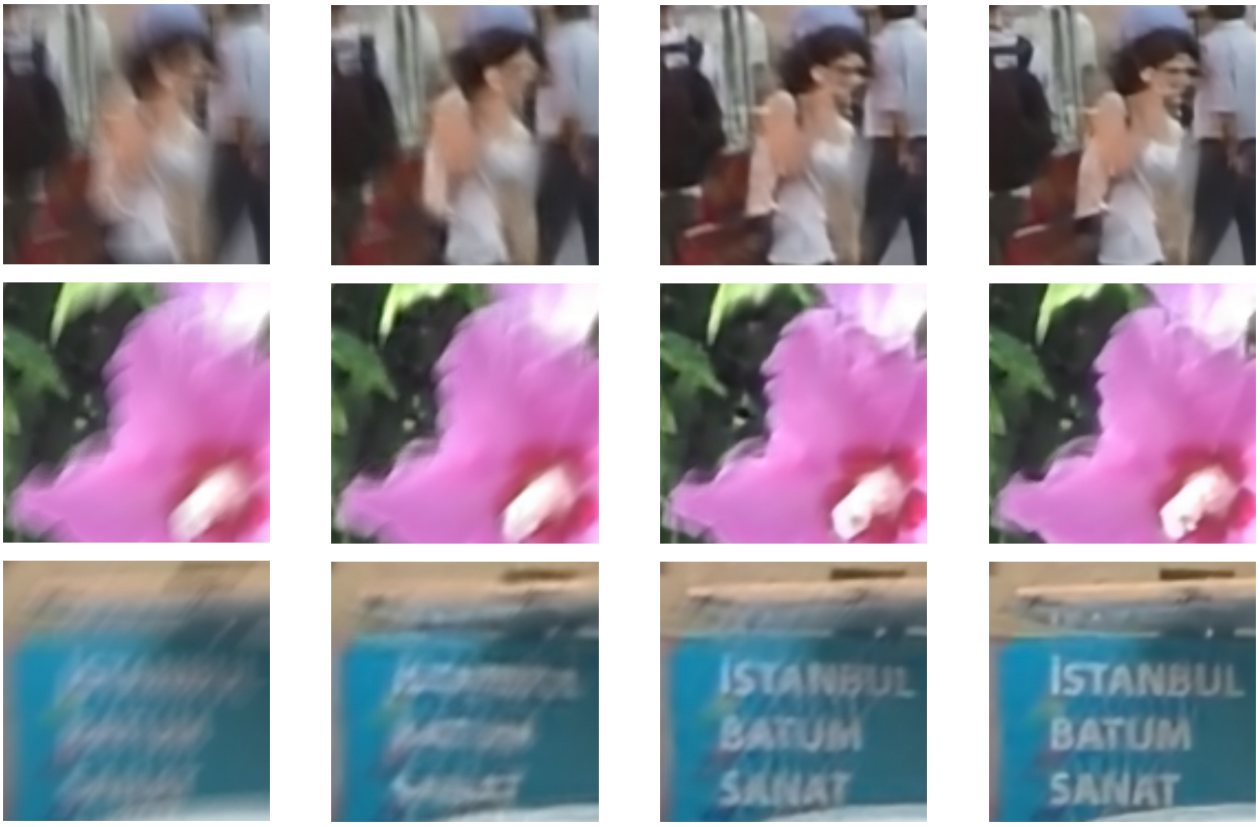
C. LOSS

The coarse-to-fine approach uses the down-sampled sharp image of the corresponding scale as the target for training and desires that each intermediate output becomes the latent sharp image in the corresponding scale. Thus, the intermediate image outputs of each iteration should form a Gaussian pyramid of sharp images and we used Euclidian loss for each scale. We want to have clear and sharp intermediate and final output images as much as possible. The loss function is defined as follows:

$$L = \sum_{j}^{M} \frac{\gamma_j}{w_j h_j c_j} \left\| I_j - I_j^* \right\|^2 \tag{2}$$

where I_j and I_j^* are the network output image and the corresponding ground truth at scale level j. The γ_j is the weight in j-th scale. We set the weights equally for each scale ($\gamma_j = 1.0$). The loss at each scale is normalized by w_j, h_j, c_j where w_j, h_j, c_j are the width, height and channels of the

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.2980996, IEEE Access



(a) Blurry image (b) $j = 1$ (c) $j = 2$ (d) $j = 3$

FIGURE 4. Progressive improvement of the intermediate output for each scale. Image scale is labelled as j, where $j = 1$ is the coarsest scalen and $j = 3$ is the finest scale. The output images become clear, progressively.

	PSNR	SSIM
single-scale network (j=1)	26.852	0.8634
multi-scale network (j=2)	28.974	0.9132
multi-scale network (j=3)	**30.612**	**0.9373**

TABLE 1. The ablation study of our network on different scales. The baseline network has residual blocks, the encoder-decoder structure and dilated convolution blocks. This table proves that the coarse-to-fine network is work obviously.

image in each scale, respectively. The M is the total scales of this network.

IV. EXPERIMENT RESULTS

We implement the deblurring network on the TensorFlow platform [26], [27] and the experiments are performed in a computer with i7-8700 CPU and NVIDIA GeForce GTX 1080 Ti GPU. Our experiments showed that our proposed method has comparable performance with state-of-the-art methods. Further details and comparisons are provided in the following sections.

A. IMPLEMENTATION DETAILS

At training, we use Adam [28] optimizer with $\beta_1 = 0.9$, $\beta_1 = 0.999$ and $\epsilon = 10^{-8}$. The learning rate are, 10^{-4} for first 1000 epochs, 10^{-5} for the 1001 to 3000 epochs, 10^{-6} for the last 3000 epochs. The initial learning rate accelerates convergence, and then it decreases gradually to fine-tune the model and increases the training stability. We set the batch size to 10.

Similar to [5], [7], our input images came from randomly cropped 256×256 patches using the images of GOPRO dataset [5] for a fair comparison. We use these patches to generate corresponding blurry and sharp image pairs. We initialize all trainable parameters by using the Xavier method [29]. Every convolution layers have the same size of kernels (5×5), and we add reflection padding to all feature maps before convolution. We found that reflection padding before convolution can improve the evaluation results on Structural Similarity (SSIM).

In this paper, we defined the number of scales j as 3, and the input image resolution as 256×256 when training. Each previous scale is quarter resolution of the succeeding scale. Specifically, the resolutions of each scale are $\{64 \times 64, 128 \times 128, 256 \times 256\}$.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.2980996, IEEE Access

(a) (b) (c) (d) (e)

FIGURE 5. This figure shows the results of our ablation study. (a) the blurry input images, (b) the output images using only residual blocks, (c) the output images are generated by our coarse-to-fine network with residual blocks and encoder-decoder structure, (d) the output images are generated by our whole network (residual blocks, dilated convolution blocks and encoder-decoder structure), and (e) the ground truth images.

	+ RB	+ ED	+ DB	PSNR	SSIM
baseline network	✓			28.138	0.8917
baseline network	✓	✓		29.466	0.9220
baseline network	✓	✓	✓	**30.612**	**0.9373**

TABLE 2. Performance with different components: the residual blocks (RB), the encoder-decoder structure (ED) and the dilated convolution block (DB). The baseline network is a coarse-to-fine architecture. We use the multi-scale j=3 in this ablation study.

	without BN (base)	with BN
PSNR	30.61	29.09
SSIM	0.9373	0.8817

TABLE 3. Comparison of image quality between our network without BN and with BN. GOPRO test dataset [5] with 1111 images were used for evaluation.

	Parameter Count	Relative
SRN [7]	6881251	×1.00
Ours	7561891	×1.10

TABLE 4. Comparison between the parameter count of our proposed method and SRN [7].

B. TRAINING AND TESTING DATASETS

To create a large dataset, the early learning–based methods used sharp images applied with uniform/non-uniform blur kernel to synthesize blurry images. However, synthetic images are still different from images captured by cameras or mobile phones.

Recently, researchers proposed GOPRO dataset [5] in which blurry images are generated by averaging consecutive short-exposure images captured by a high-speed camera. The generated images are realistic because it simulates complex movements like camera shake and object motions, which are common in a real-world setting.

To compare with other networks, we train our network with the GOPRO dataset, similar with [5], [7]. The GOPRO dataset consists of 3,214 image pairs (blurry/sharp). Following the same strategy as [5], [7], we used 2,103 images pairs for training and the remaining 1,111 image pairs for testing.

We showcase the image results of the different deblurring methods using *Real image dataset* [30], which contains images for real-world scenarios. This dataset contains non-synthetic blurry images that were captured "in the wild".

C. ABLATION STUDY

We examined the model to see its performance using different configurations. We examined the effectiveness of the coarse-to-fine scheme by checking the deblurring result for each scale. The results of the experiment could be seen in Table 1 and Fig 4. In Fig 4, we could see that the output images become clearer, progressively.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.2980996, IEEE Access

Methods	Kim et al. [12]	Whyte et al. [4]	Xu et al. [31]	Sun et al. [18]	MSCNN [5]	SVRNN [6]	SRN [7]	Ramakrishnan et al. [32]	Kupyn et al. [19]	Ours
PSNR	23.64	23.53	25.18	24.64	29.13	29.25	30.20	28.94	28.70	**30.61**
SSIM	0.8239	0.8458	0.8960	0.8429	0.9111	0.9189	0.9317	0.9220	0.858	**0.9373**
Time	50min	8.5min	50min	17min	2.63s	1.19s	1.59s	-	0.72s	**0.66s**

TABLE 5. The comparison of the proposed method with other approaches. We average PSNR, SSIM of 1111 images on GOPRO testing dataset [5], and we also test the running time. The value highlighted in bold type has the best performance.

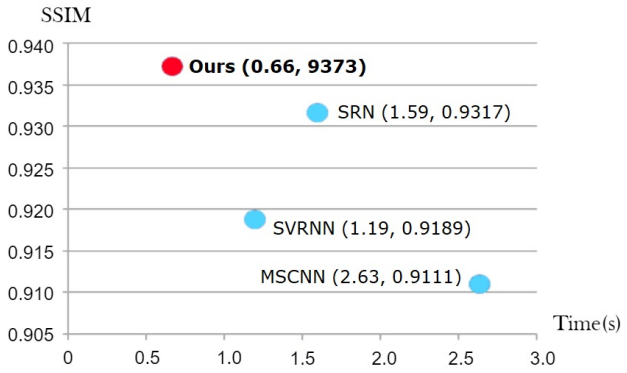


FIGURE 6. Performance of our method compared to state-of-the-art approaches (MSCNN [5], SVRNN [6], SRN [7]). The execution time of the deblurring process is defined in the x-axis and is measured using seconds (s). The image quality of the deblurring result is defined in the y-axis and is measured using Structural Similarity (SSIM). The fastest execution time is in the leftmost and the best image quality is in the topmost. Our proposed method outperforms other state-of-the-art deblurring approaches in terms of both execution time and image quality.

We also analyzed the effect of each major components of the proposed network G_θ, which are residual blocks, the encoder-decoder structure, and the dilated convolution block. The results of this experiment could be seen in Table 2 and Fig 5. When including all of the components, the network provides the highest performance. Fig 5 (b) shows the output image using only residual blocks. We can see that the outputs could already capture the overall structures of the images. With the encoder-decoder structure, the output images are much sharper and the object boundaries are further refined to follow the original shapes of the objects, as shown in Fig 5 (c). Incorporating the dilated convolution blocks further sharpen the resulting images. This is most noticeable in the second row of Fig 5 (d) where the texts are sharper than the previous results.

Furthermore, we examined the effects of using batch normalization (BN) in our network. In Table 3, our results show that the network without BN has better performance than the network with BN. BN also increases the memory usage of the network since BN layers consume similar memory as the previous convolution layer.

We also compare the number of parameters between our proposed method and our baseline, SRN [7]. In Table 4, the number of parameters of our model is greater than the baseline SRN by around 10%. However, we achieved a lower execution time because our architecture allows parallel computation.

D. COMPARISON WITH STATE-OF-THE-ART METHODS

We compare the proposed method with the state-of-the-art deblurring approaches using the GOPRO dataset [5], in terms of image quality and execution time. We used the metrics, Peak Signal-to-Noise Ratio (PSNR) and Structural Similarity (SSIM), to measure the image quality of the deblurred image. The comparison results are listed in Table 5. Figure 6 contains the visualization of the comparison between the proposed method and the state-of-the-art methods.

Based on our results found in Figure 6, our proposed network performs favorably against state-of-the-art blind motion deblurring approaches that use deep learning (i.e. MSCNN. [5], SVRNN [6], SRN [7]). The deblurring result of has higher image quality compared to others, as measured by SSIM. Furthermore, the execution time of our method is at least twice faster compared to other methods.

The earlier methods, Kim et al. [12] and Sun et al. [18], are unsuccessful at nonlinear blur region or motion boundary because strong edges are not found. Although Sun et al. [18] used CNN, they used it to predict kernel direction. In most cases, their methods cannot predict a reliable kernel. The recent learning-based methods [5], [7], [19] have significantly better performance compared to earlier methods, in terms of image quality and execution time. We used the publicly available implementation of the algorithms to get the benchmark. The proposed method effectively produces better results with clearer edges and more details. As shown in Table 5, the PSNR of our method higher 2 dB than Kupyn et al. [19], and the SSIM of our method is also 0.2 higher than MSCNN [5] and SVRNN [6]. Our execution time is much faster as compared to previous learning-based methods. It is two times faster than SRN [7]. Based on the results, our model showed improved image quality and faster running time. The examples of output images are shown in Fig 7. The picture below is the zoomed-in patch marked by the red box in the above picture.

We compared our results to recent learning-based works on real-world dataset [30]. MSCNN [5] and SRN [7] both demonstrated good results, but sometimes there are ringing visual artifacts. Our proposed model can restore blur, caused by complex or large movement, with excellent results and less visual artifacts. In Fig 8, our proposed method effectively produces better results with clearer edges and details. At the first row in Fig 8, the blurry image is the picture of the Roman Colosseum, and the image has severe blurs that the deblurring methods cannot recover a sharp image. Our proposed method can recover sharper details as seen in the sample image.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.2980996, IEEE Access

(a) blurry image (21.992dB, 0.7192) (b) MSCNN [5] (24.565dB, 0.8199) (c) SRN [7] (26.095dB, 0.8740) (d) Ours (26.406dB, 0.8808)

(e) blurry image (26.885dB, 0.9051) (f) Nah et al. [5] (31.513dB, 0.9497) (g) SRN [7] (33.159dB, 0.9644) (h) Ours (33.308dB, 0.9651)

(i) blurry image (28.010dB, 0.9357) (j) MSCNN [5] (31.763dB, 0.9590) (k) SRN [7] (32.514dB, 0.9661) (l) Ours (32.814dB, 0.9676)

FIGURE 7. Testing on the GOPRO dataset [5]. This figure shows the results with PSNR and SSIM values of the various methods for image deblurring and the picture below is the zoomed-in patch marked by the red box in the above picture. In the first row, our result has the strongest edge after deblurring. In the second and third rows, our details on the man face are more than others, and the boundary convergence of the earphone cable is the best in our image.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.2980996, IEEE Access

FIGURE 8. Deblurring results using some images from *Real image dataset* [30]. The picture below is the zoomed-in patch marked by the red box in the above picture. In the first row, the blur of the image is too severe to recover details. Our proposed method generates a more reasonable image compared to others. In the second row, the result of MSCNN [5] has a lot ringing visual artifacts and the result of SRN [7] is still blurry. Our method produces the least blurry output. In the final row, we have a significantly better deblurring result which has stronger edges without ringing visual artifacts, and more readable text.

At the second row, the input image is a portrait of a person and we focus in the facial features. The noise and ringing visual artifacts of our output images are less than the other output of the other methods. For MSCNN [5] their output is still blurry while SRN [7] and ours have a clearer deblurred face. Our method produced a less blurry eyes compared with SRN [7]. At the third row, our focus is in the deblurred text and we have a significantly better deblurring result. The

reconstructed text of our method has fewer ringing artifacts and is more readable than the other methods. For MSCNN [5] and SRN [7], there are still ringing artifacts. Overall, Our deblurred image has stronger edges with fewer ringing visual artifacts, and more readable text.

V. CONCLUSION

In this paper, we proposed an end-to-end image deblurring method that deblur images with non-uniform blurs using U-Net architecture and multi-scale features extracted using coarse-to-fine scheme and aggregated dilated convolution. The two components of our method (i.e. coarse-to-fine scheme, aggregated dilated convolution) can extract multi-scale features by themselves; however, combining the two methods have a multiplicative effect by extracting multi-scale features (i.e. aggregated dilated convolutions) at different scales of the images (i.e. coarse-to-fine scheme). Our method has better performance compared to the state-of-the-art methods in terms of image quality, characterized by less ringing artifacts found in our output images. Furthermore, our network uses parallel optimization to decrease the execution time of the deblurring process.

VI. ACKNOWLEDGEMENT

This work was financially supported by the Center for Cyber-physical System Innovation and Center of Intelligent Robots from The Featured Areas Research Center Program within the framework of the Higher Education Sprout Project by the Ministry of Education (MOE) in Taiwan and Ministry of Science and Technology of Taiwan under Grants MOST108-2218-E-011-026, MOST108-2221-E-011-116, MOST108-2622-E-011-016-CC3.

REFERENCES

[1] A. Levin, Y. Weiss, F. Durand, and W. T. Freeman, "Understanding and evaluating blind deconvolution algorithms," in IEEE Conference on Computer Vision and Pattern Recognition (CVPR). IEEE, 2009, pp. 1964–1971.

[2] D. Gong, J. Yang, L. Liu, Y. Zhang, I. Reid, C. Shen, A. V. D. Hengel, and Q. Shi, "From motion blur to motion flow: a deep learning solution for removing heterogeneous motion blur," in IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017, pp. 2319–2328.

[3] A. Gupta, N. Joshi, C. L. Zitnick, M. Cohen, and B. Curless, "Single image deblurring using motion density functions," in European Conference on Computer Vision (ECCV), 2010, pp. 171–184.

[4] O. Whyte, J. Sivic, A. Zisserman, and J. Ponce, "Non-uniform deblurring for shaken images," International Journal of Computer Vision (IJCV), pp. 168–186, 2012.

[5] S. Nah, T. H. Kim, and K. M. Lee, "Deep multi-scale convolutional neural network for dynamic scene deblurring," in IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017, pp. 3883–3891.

[6] J. Zhang, J. Pan, J. Ren, Y. Song, L. Bao, R. W. Lau, and M. H. Yang, "Dynamic scene deblurring using spatially variant recurrent neural networks," in IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2018, pp. 2521–2529.

[7] X. Tao, H. Gao, X. Shen, J. Wang, and J. Jia, "Scale-recurrent network for deep image deblurring," in IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2018, pp. 8174–8182.

[8] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI), 2015, pp. 234–241.

[9] C. J. Schuler, M. Hirsch, S. Harmeling, and B. Schölkopf, "Learning to deblur," IEEE transactions on pattern analysis and machine intelligence, vol. 38, no. 7, pp. 1439–1451, 2015.

[10] F. Yu and V. Koltun, "Multi-scale context aggregation by dilated convolutions," International Conference on Learning Representations (ICLR), 2016.

[11] H. Miao, W. Zhang, and J. Bai, "Aggregated dilated convolutions for efficient motion deblurring," in IEEE International Conference on Multimedia and Expo (ICME), 2018, pp. 1–6.

[12] T. H. Kim and K. M. Lee, "Segmentation-free dynamic scene deblurring," in IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2014, pp. 2766–2773.

[13] J. Pan, Z. Hu, Z. Su, and M.-H. Yang, "Deblurring text images via l0-regularized intensity and gradient prior," in IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2014, pp. 2901–2908.

[14] J. Pan, D. Sun, H. Pfister, and M.-H. Yang, "Blind image deblurring using dark channel prior," in IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 1628–1636.

[15] Y. Bahat, N. Efrat, and M. Irani, "Non-uniform blind deblurring by reblurring," in IEEE International Conference on Computer Vision (ICCV), 2017, pp. 3286–3294.

[16] C. Cai, H. Meng, and Q. Zhu, "Blind deconvolution for image deblurring based on edge enhancement and noise suppression," IEEE Access, vol. 6, pp. 58 710–58 718, 2018.

[17] X. Mao, C. Shen, and Y. B. Yang, "Image restoration using very deep convolutional encoder-decoder networks with symmetric skip connections," in Advances in Neural Information Processing Systems (NIPS), 2016, pp. 2802–2810.

[18] J. Sun, W. Cao, Z. Xu, and J. Ponce, "Learning a convolutional neural network for non-uniform motion blur removal," in IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2015, pp. 769–777.

[19] O. Kupyn, V. Budzan, M. Mykhailych, D. Mishkin, and J. Matas, "Deblurgan: Blind motion deblurring using conditional adversarial networks," in IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2018, pp. 8183–8192.

[20] L. Li, J. Pan, W.-S. Lai, C. Gao, N. Sang, and M.-H. Yang, "Blind image deblurring via deep discriminative priors," International Journal of Computer Vision, pp. 1–19, 2019.

[21] J. Johnson, A. Alahi, and L. Fei-Fei, "Perceptual losses for real-time style transfer and super-resolution," in European conference on computer vision. Springer, 2016, pp. 694–711.

[22] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 770–778.

[23] B. Lim, S. Son, H. Kim, S. Nah, and K. M. Lee, "Enhanced deep residual networks for single image super-resolution," in IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), 2017, pp. 136–144.

[24] K. He, X. Zhang, S. Ren, and J. Sun, "Identity Mappings in Deep Residual Networks," Welcome to the Computing Research Repository (CoRR), 2016.

[25] ——, "Deep residual learning for image recognition," in IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 770–778.

[26] S. S. Girija, "Tensorflow: Large-scale machine learning on heterogeneous distributed systems," Software available from tensorflow. org, 2016.

[27] M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard et al., "Tensorflow: A system for large-scale machine learning," in USENIX Symposium on Operating Systems Design and Implementation (OSDI), 2016, pp. 265–283.

[28] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," International Conference on Learning Representations (ICLR), 2015.

[29] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," in International Conference on Artificial Intelligence and statistics (AIStats), 2010, pp. 249–256.

[30] W. S. Lai, J. B. Huang, Z. Hu, N. Ahuja, and M. H. Yang, "A comparative study for single image blind deblurring," in IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 1701–1709.

[31] L. Xu, S. Zheng, and J. Jia, "Unnatural l0 sparse representation for natural image deblurring," in IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2013, pp. 1107–1114.

[32] S. Ramakrishnan, S. Pachori, A. Gangopadhyay, and S. Raman, "Deep generative filter for motion deblurring," in IEEE International Conference on Computer Vision (ICCV), 2017, pp. 2993–3000.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.2980996, IEEE Access

JOSE JAENA MARI OPLE received his B.S. degree in Computer Science from De La Salle University, Philippines, in 2018. He is currently pursuing M.S. degree in Computer Science at National Taiwan University of Science and Technology. His research interests include digital image processing and deep learning applied to computer vision.

PIN-YI YEH received the B.Eng. degree in computer science information engineering from Fu Jen Catholic University in 2017. She is currently pursuing the M.Sc. degree with the Department of Computer Science and Information Engineering, National Taiwan University of Science and Technology. Her research interests include digital image processing and deep learning applied to computer vision.

SHIH-WEI SUN received the B.S. degree from Yuan-Ze University and Ph.D. degree from National Central University, Taiwan, in 2001 and 2007, respectively, both in Electrical Engineering. From 2007 to 2012, he was a post doctoral research fellow at the institute of information science, Academia Sinica. In 2012, he joined the Department of New Media Art, Taipei National University of the Arts, Taiwan, as an assistant professor. Since 2016, he is an associate professor. He is the founding leader with the Ultra-Communication Vision Laboratory (ucVision Lab). His research interest includes: computer vision, sensor applications for mobile devices, and applications for human-computer interface. He received the Prize Award of Multimedia Grand Challenge from the ACM Multimedia Conference (MM) in 2014, and the Excellent Poster Award from the IEEE Global Conference on Consumer Electronics (GCCE) in 2018. He published more than 50 international journal papers and conference papers. He serves as the reviewers and technical program committee members for many international journals and academic conferences. In addition, Dr. Sun advised the technical production for the tech-art installations and performances, exhibited in Taipei Fine Art Museum and Songshan Cultural and Creative Park, which are the most representative exhibition places in Taiwan, in 2013 and 2015.

I-TE TSAI received the B.Eng. degree in computer science from National Taiwan University of Science and Technology. He is currently pursuing the M.Sc. degree with the Department of Computer Science and Information Engineering, National Taiwan University of Science and Technology. His research interests include digital image processing and deep learning applied to computer vision.

KAI-LUNG HUA received the B.S. degree in electrical engineering from National Tsing Hua University in 2000, and the M.S. degree in communication engineering from National Chiao Tung University in 2002, both in Hsinchu, Taiwan. He received the Ph.D. degree from the School of Electrical and Computer Engineering, Purdue University, West Lafayette, IN, in 2010. Since 2010, Dr. Hua has been with National Taiwan University of Science and Technology, where he is currently a professor in the Department of Computer Science and Information Engineering. He is a member of Eta Kappa Nu and Phi Tau Phi, as well as a recipient of MediaTek Doctoral Fellowship. His current research interests include digital image and video processing, computer vision, and machine learning. He has received several research awards, including 2019 Outstanding Research Award of Taiwan Tech, 2018 Young Scholar Award of Taiwan Tech, Top Performance Award of 2017 ACM Multimedia Grand Challenges, Top 10% Paper Award of 2015 IEEE International Workshop on Multimedia Signal Processing, the Second Award of the 2014 ACM Multimedia Grand Challenge, the Best Paper Award of the 2013 IEEE International Symposium on Consumer Electronics, and the Best Poster Paper Award of the 2012 International Conference on 3D Systems and Applications.