Attention! Stay Focus!

Tu Vo
BridgeAI Inc.
Seoul, South Korea
tuvovan@pukyong.ac.kr

Abstract

We develop a deep convolutional neural networks (CNNs) to deal with the blurry artifacts caused by the defocus of the camera using dual-pixel images. Specifically, we develop a double attention network which consists of attentional encoders, triple locals and global local modules to effectively extract useful information from each image in the dual-pixels and select the useful information from each image and synthesize the final output image. We demonstrate the effectiveness of the proposed deblurring algorithm in terms of both qualitative and quantitative aspects by evaluating on the test set in the NTIRE 2021 Defocus Deblurring using Dual-pixel Images Challenge [1] [4]. The code, and trained models are available at https://github.com/tuvovan/ATTSF.

1. Introduction

In general, the exposure of the image can be adjusted by two different ways. The first way is to change the shutter speed when the aperture is fixed to control the amount of light falling on the sensor. The other way is to keep the shutter speed unchanged while adjusting the aperture's size. While the former method may cause the motion blur if there is any possible object motion, the later method results in a shallow depth of field (DoF), causing defocus blur to occur in scene regions outside the DoF [3] . Removing the defocus blur is critical as we can obtain an image which is captured using a wide aperture but still have everything in focus, which ensures a well-exposed image with sufficiently sharp image.

In theory, defocus blur is a result of a sharp region with a spatial point spread function (PSF) that use the neighborhood pixel in producing the blurry pixel [15]. As a result, using the dual-pixel alone may not sufficiently enough to faithfully recover the original sharp pixel. However, we believe that by employing the large receptive fields provided by stacking convolution layers with maxpooling, a neural network will be able to produce the non-blurred outputs, given the dual-pixel inputs. Recently, Abuolaim *et al.* [3] trained an Unet-like model which takes two images as input and produces a defocus blur-free output. Unet is an encoder-decoder framework which considers all pixel and channel equally which we think not suitable as the blurry pixels are distributed differently for each channel and each pixel location.

In this work, to employ different feature in both input images intentionally, we propose an attention deep convolutional neural networks (CNNs) to remove the defocus blur artifact which is built upon Encoder - Decoder architecture with the Dual Attention Modules. As mentioned above, we notice that every pixel and channel of the input images should be considered appropriately, make them contribute to the final output at different level. As a result, we redesigned the encoder to extract the useful information by adding the dual-attention module to the classical encoder module. Furthermore, the extracted features from the attention-encoder will be concatenated and put through the triple-local and global-non-local modules before being decoded by decoder modules and generate the sharp output image. We demonstrate the effectiveness of the proposed network through the *NTIRE2021 Defocus Deblurring Challenge [1] [4]*. Using the data provided by the competition, we trained a network and finally archived the average *PSNR* of 26.4243 *dB*, stands at the 9^{th} position in the competition.

2. Related Works

2.1. Defocus Deblurring

While there are many previous works on deblurring field, we found that those methods that try to estimate the defocus map and deblurring are the closest methods as they all try to produce the sharp and deblurred output. The most common method for defocus deblurring task is to first estimate the deblurring kernel and then use that kernel as a guidance for deblurring. To find the deblurring kernel, Park *et al.* [13] fed a combination of pretrained blur classification network to extract the deep blur feature along with the hand-crafted feature to a regression network to estimate the amount of

blur in the pixel edge to later deblur it. Karaali *et al.* [9] extracted the difference between gradient of the blurry image and the original one. And most recently, Abuolaim *et al.* [3] introduced a deep learning model, which consists of encoder and decoder modules, use the dual-pixel data to directly solve for the defocus blur in a single step without estimating the defocus map.

2.2. Attention Modules in Deep Learning

Recently, attention mechanisms show the effectiveness in many computer vision fields including the image restoration. Thanks to its ability to facilitate deep neural networks to determine where to focus and improve the representation of interest [16]. By observing that each pixel and each image channel should be considered separately, we end up adding the dual-attention [7] module to the conventional encoder in our proposed network to tackle with the defocus deblurring challenge. By incorporating the attention mechanisms with the conventional encoder modules, every pixel and channel is dealt separately, make sure they contribute the useful information at different level before being merged and being decoded. With this mechanism, the proposed architecture yields high-quality results in both qualitative and quantitative perspective.

2.3. Defocus Blur Dataset

There are several available datasets for the defocus deblurring task. Salvado *et al.* [6] proposed the RTF dataset which has 22 image pairs of blurred image and its corresponding in-focus-image. The CUHK [14] and DUT [18] provide real-blurred images with the corresponding binary masks representing the blur/sharp regions. This dataset is more suitable for blur detection task. Recently, Abuolaim *et al.* [3] proposed a large dataset with 500 different pairs of non-overlapped scenes. By using the dual-pixels, the dataset is extended to 2000 images with blur images and the corresponding sharp images. This dataset is used for *NTIRE2021 Defocus Deblurring Challenge [1] [4]*.

3. Attention! Stay Focus! (ATTSF)

We design an attention encoder decoder network to effectively synthesize the blurry input images and generate a high-quality blur-free output. Figure 1 shows the architecture of the proposed network, which takes two images (left blurry image and right blurry image) as input and then reconstructs a sharp image. In details, our proposed ATTSF consists of several attention encoders, triple local, global-local blocks and decoder modules. The attention encoders are used to extract useful features from the blurry input images. The features generated from those encoders are concatenated together and being transferred to the triple-local and global non-local modules in parallel then finally being decoded to get the final sharp output image. To ensure the output image to have the useful feature from the input images, we use the skip-connection to connect the output feature of the encoders and decoders at every level.

3.1. Attention Encoder (ATTE)

The conventional encoder usually consists of several convolution layers following by the activation layers and pooling layers. Encoder modules are good at extracting the high-level feature of the input image. However, all pixel and channel are treated equivalently which is not strong enough in this defocus deblurring challenge in our opinion. We observe that the defocus deblurring challenge, the blur level are not equally distributed both over image channels and image pixel. As a results, we employ the dual-attention mechanism, which is composed of channel attention and pixel attention or position attention. Figure 2 shows the architecture of the dual-attention modules. To be specific, each ATTE block has its input to go through the dual-attention module to extract the high-level feature intentionally. Follow the dual-attention module is a couple of convolution layers with ReLU [12] activation functions and MaxPooling layers, just similar to the conventional encoder.

Channel Attention

As the input of the network is dual-pixel, each image should contribute different kind of information to the final output image. Base on this we employed the dual-attention module which includes the channel attention and pixel attention. The channel attention intentionally extract the feature across the channel dimension by calculating the channel attention map from the input feature. The channel attention applies the convolution layer following by the sigmoid function, which ensures that the attention map will range from 0.0 to 1.0, representing the amount of the information contribution of each feature channel to the output. By masking the input feature with the calculated attention mask, we get the output feature which consists of useful information from each channel of the input.

Pixel Attention

In addition to channel attention, pixel attention a.k.a position attention is also crucial for this task. While channel attention works on channel domain, pixel attention, on the other hand, pays attention to the every pixel in the input feature. The module applies global average pooling(GAP) and global max pooling(GMP) in parallel on the input feature. The GAP and GMP feature are then concatenated together, follows by a 1×1 convolution with the sigmoid activation function to generate a pixel attention mask. The attention mask is then multiply equally with every input channel, resulting in the output feature.

Dual Attention

Having the pixel attention and channel attention, the dual attention module takes the input feature and applies two 3×3 convolution layers follows by ReLU [12] activation

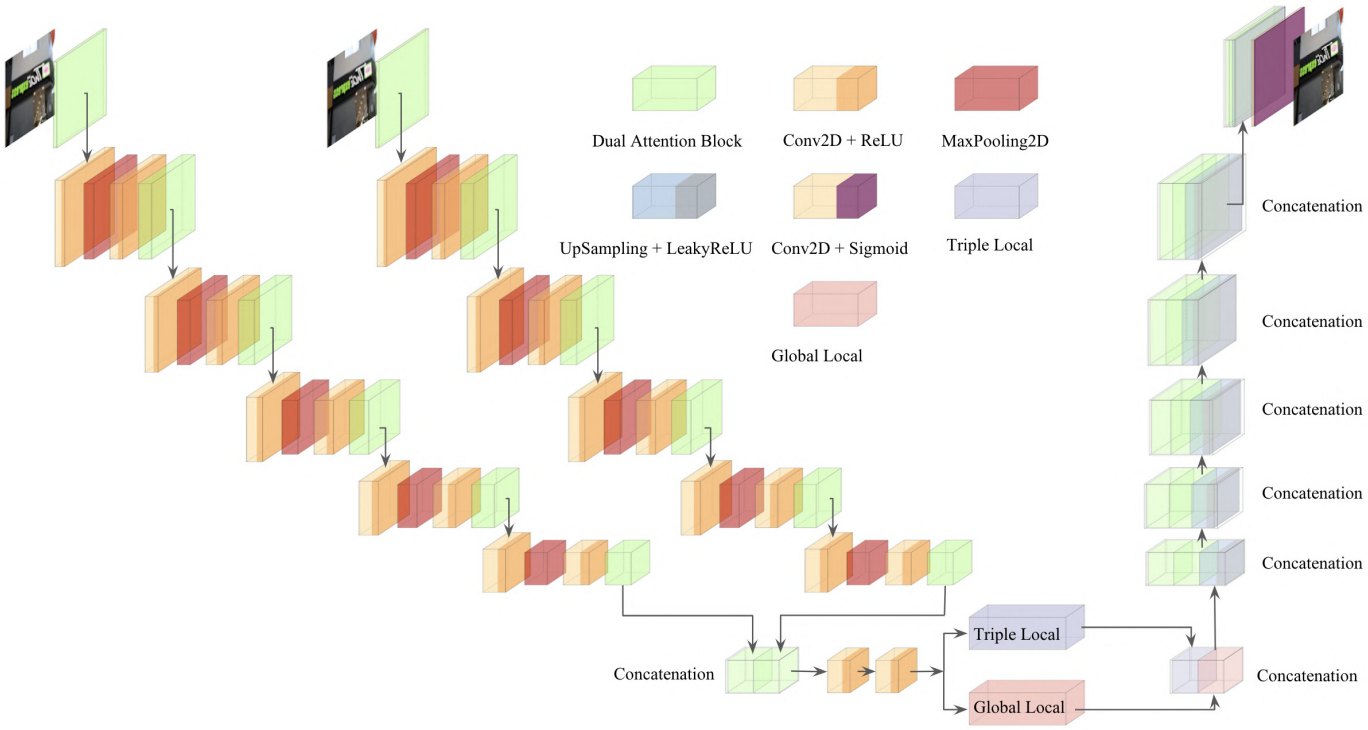


Figure 1: The overall architecture of the proposed ATTSF.

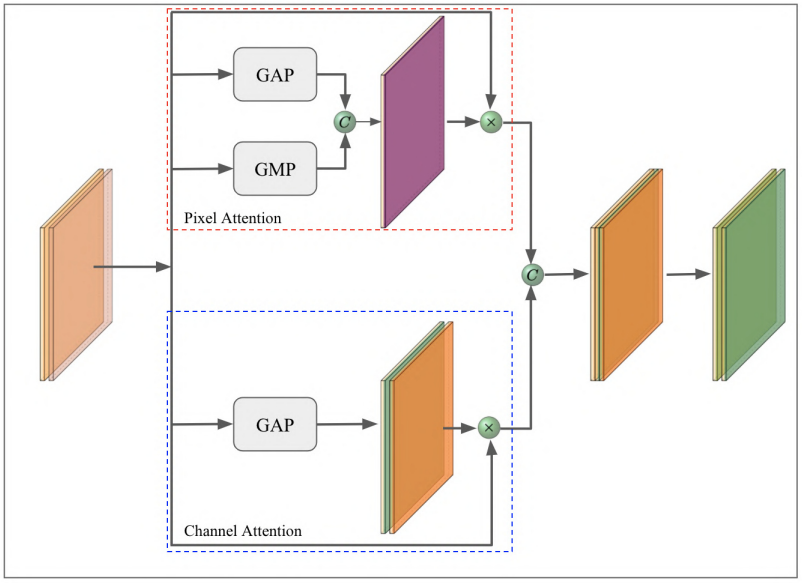


Figure 2: Dual Attention Module. GAP, GMP are Global Average Pooling and Global Max Pooling, respectively. $\times$ denotes channel-wise multiplication, and C denotes the concatenate operation.

function. The feature is then put through Pixel Attention and Channel Attention simultaneously and being concatenated in the channel axis. Finally, the concatenated feature is 1×1 convoluted to match the dimension of the input fea-

ture, as shown in Figure 2.

Attention Encoder

As mentioned before, the proposed attention encoder is built based on the conventional encoder scheme by adding the dual attention module on top of it. Specifically, the input feature first goes through the dual attention module, then through the encoder part, which is composed of several 3×3 convolution layers and ReLU activation functions [12].

3.2. Triple Local

Figure 3 shows the architecture of the triple local module. This is inspired by the inception modules, which has multiple convolution kernel with different size, in order to extract the feature of different levels. The small filter is able to extract local details of the features, and the large filter can cover larger regions of the receiving layers. All the features are concatenated in a channel-wise manner and compressed through a convolutional layer.

3.3. Global Local

Figure 4 illustrates the architecture of the global local module. As we know, the convolution represents the local feature. In this task, although the local features are essential, we do not want to loose the global terms as it makes the whole image to be spatially consistent. Here we employed the idea from [17] and [5] which calculate the correlation between two input signals of the whole image. The global local module cover large receptive fields so the network can ensure the spatial consistency, avoiding the hallucination.

3.4. Implementation Details

In our implementation, each convolution in encoder module is followed by a Rectifier Linear Unit (ReLU) activation function [12], while each layer in decoder module is followed by a Leaky Rectifier Linear Unit (Leaky ReLU) [11]. The reason behind using the Leaky ReLU instead of ReLU is to avoid the under-bound of the hidden layer's output, which may lead to unwanted reconstructed image. Every layer is initialized follow the *He normal* [8], and all convolution kernel in encoders or decoders are 3×3 . In each training batch, we apply several augmentation technique such as random rotation, horizontal and vertical flipping. All input images are normalized between 0.0 and 1.0.

We first trained the networks using the Adam optimizer [10] with the learning rate of 1×10^{-4}, and the batch size was set to 4 for 200 epochs. We then change the loss function to loss function 2 as shown in Eq. 1 and train the model with SGD optimizer with the batch size of 2, learning rate of 5×10^{-5} with 100 more epochs, where $\alpha = 1$, $\beta = 0.5$, respectively. We also apply the Learning Rate Scheduler to decrease the learning rate by half every 20 epochs. Finally, the model is implemented using Tensorflow [2] with the help of Tensorflow Addons package. We

Method	*PSNR*	*SSIM*	*MAE*
Abuolaim *et al.* [3]	25.13	78.59	0.0406
Ours	**25.98**	**81.15**	**0.0377**

Table 1: *PSNR*, *SSIM* and *MAE* values of our proposed algorithm, compared with Abuolaim *et al.* [3] The bold values indicates the better results. We can notice that our network is far better than the state of the arts.

experimentally found that finetuning the network again with the loss function 1 that has more weight on *SSIM* , the model not only gets a high *PSNR* but also high *SSIM* value. And by achieving high *SSIM* score, the final predicted images are more similar to the ground-truths, make them become closer to the real sharp images.

We used the training dataset provided by the *NTIRE2021 Defocus Deblurring Challenge [1] [4]*. The dataset is divided into three parts: training, validation and testing . We used the training set for training and validation set, test set for validating and testing the model, respectively. Because of the memory limitation, we did not use the original images for training, but we cropped into many patches of size 560×560 from the training and validation sets by sliding over images with strides of 140×140. As a results, we end up with more than 2000 images for training and 500 images for validation. For testing, we keep the original sizes. The training took approximately three days using a computer with Intel® Core™ i7, 32GB RAM, and Nvidia V100 GPU. After training and fine-tuning, we use the original test set provided by the competition. The model takes about 0.5 second for one image, which is reasonably fast.

$$Loss = \alpha \times SSIMLoss + \beta \times MAELoss \quad (1)$$

4. Experimental Results

4.1. Quantitative and Qualitative Evaluation

We evaluate the performance of the proposed network using the test set provided by the *NTIRE2021 Defocus Deblurring Challenge [1] [4]*. The competition provides two different test sets. One of them has ground truth images which we use to compare the qualitative metrics with state-of-the-art methods recently, the other one does not include the ground truth, so we only use them for visual comparison, as shown on Figure 7.

Table 1 compares the *PSNR*, *MAE* and *SSIM* calculated using the former test set mention above. We only compare with Abuolaim *et al.* [3] method as we notice that Abuolaim *et al.* [3] 's method is the only method that tries to solve the defocus deblurring problem using deep learning model, which is close to our work. The proposed method outper-

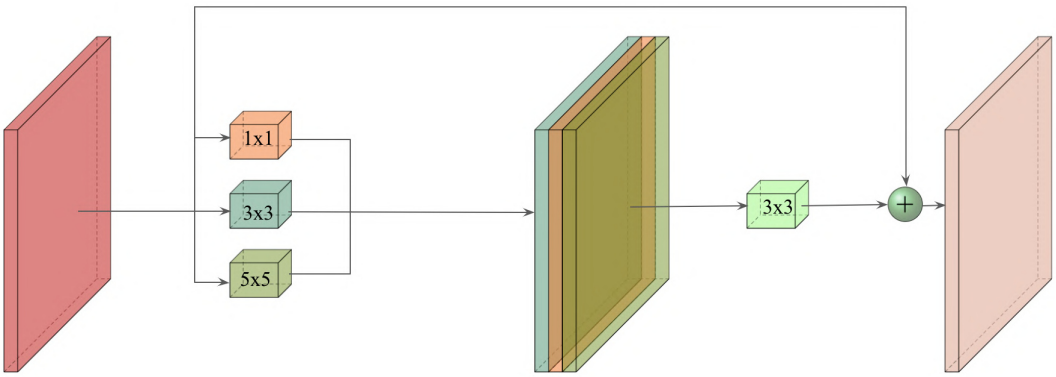


Figure 3: Triple Local Module

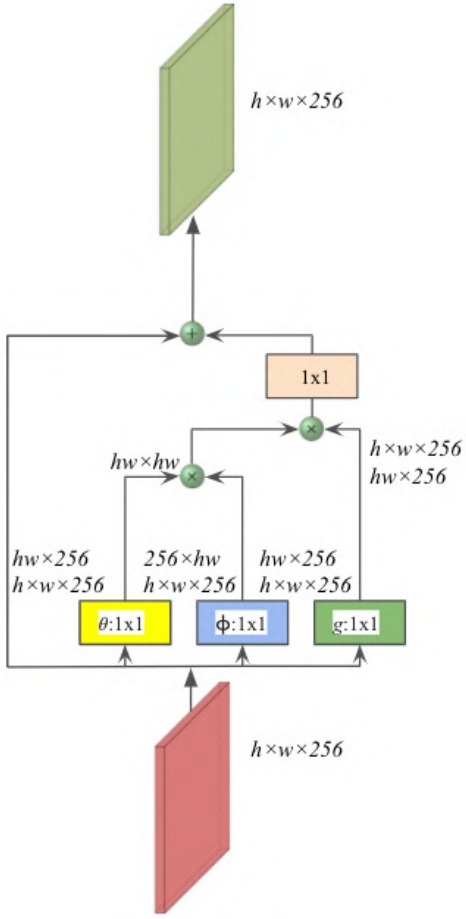
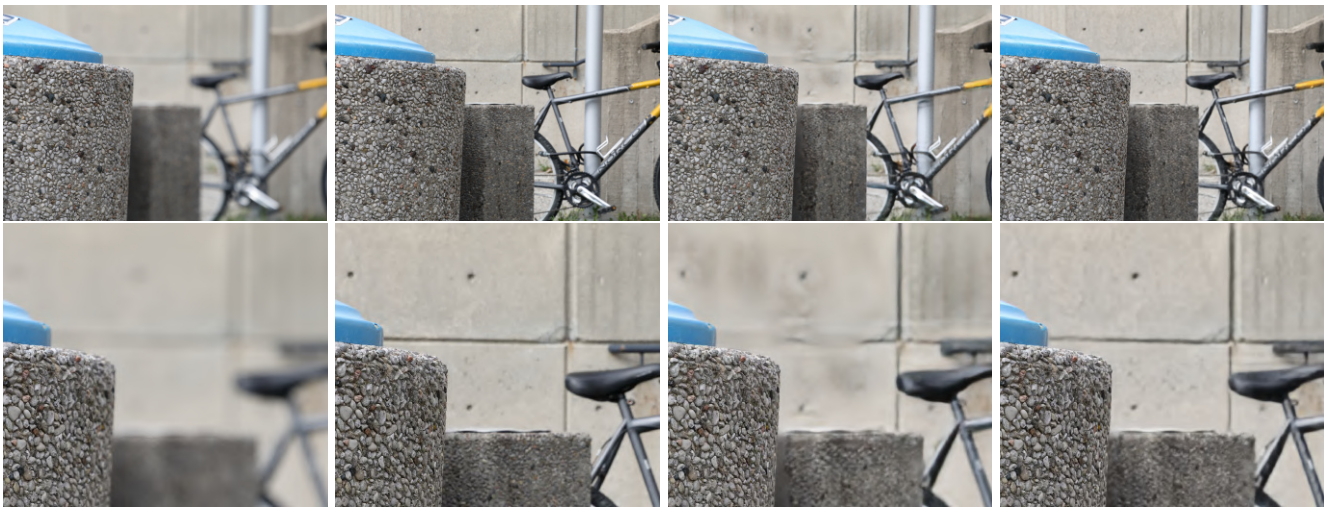


Figure 4: Global Local Module

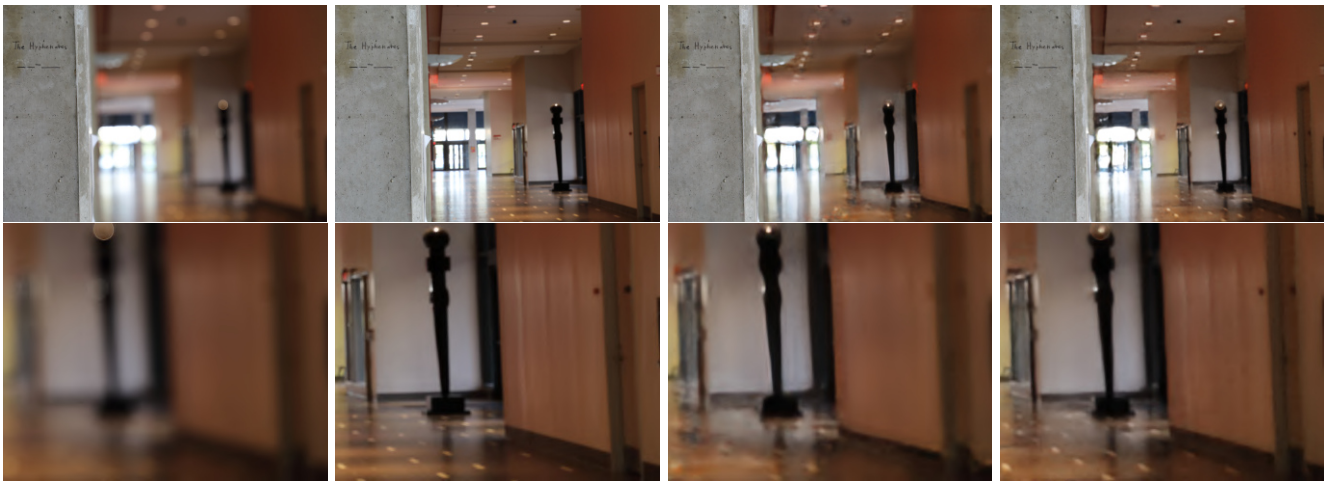
forms the state-of-the-art algorithm in terms of qualitative metrics, as the feature of the input images are extracted attentionally, and contributed to output the sharp images.

Figure 5 and Figure 6 visually compare the defocus deblurring results on the test set with ground truths provided. Abuolaim *et al.* [3] still yields blur artifacts while ATTSF preserves sharp edges and fine details more faithfully. We also verify the effectiveness of the our proposed method us-



(a) Input (b) Ground Truth (c) Abuolaim *et al.* [3] (d) Proposed

Figure 5: Visual comparison of the proposed algorithm and Abuolaim *et al.* [3] 's algorithm. The proposed algorithm outperforms the Abuolaim *et al.* [3] as it faithfully recovers the wall region, makes it close to the ground truth image.



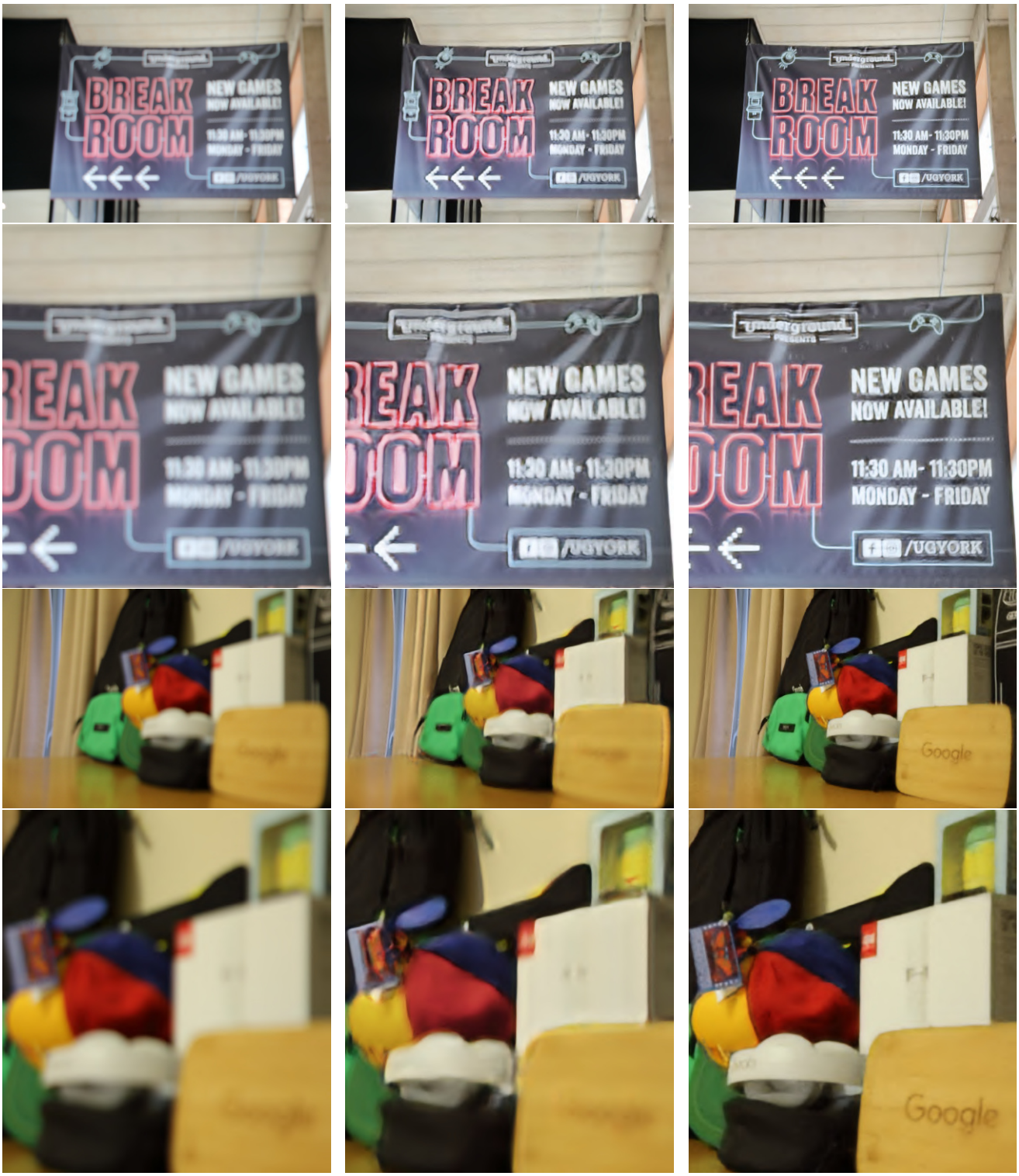
(a) Input (b) Ground Truth (c) Abuolaim *et al.* [3] (d) Proposed

Figure 6: Visual comparison of the proposed algorithm and Abuolaim *et al.* [3]'s algorithm. The proposed algorithm successfully deblur the blurry regions such as the door or the light on the ceiling.

ing the second test set provided by the competition. As this set does not contain the ground truth images, we only able to compare our results without the ground truth. The results shown on Figure 7 again show that our proposed method successfully recover the blur region and out-perform the state-of-the-art algorithm. Although there was no ground truth to compare the results qualitatively, it is reported that the our *PSNR* on this test set was 26.4243 *dB*, 9^{th} position in the competition.

5. Conclusion

In this work, we proposed an attention deep learning network which leverages the original encoder and decoder architecture by adding the dual-attention modules before every encoder blocks to attentionally extract the feature in each blur input image. Furthermore, at the bottleneck point, we also added the triple local and global local modules in parallel to efficiently extract the local features in different level as well as keep the global context of the input images. The features are then being concatenated with the encoded feature at every level and being decoded by the decoder



(a) Input (b) Abuolaim *et al.* [3] (c) Proposed

Figure 7: Visual comparison of the proposed algorithm and Abuolaim *et al.* [3]'s algorithm. Abuolaim *et al.* [3] fails to produce the non-blur output while the proposed algorithm faithfully remove the blur artifact and generate sharp images.

modules, then finally restore the sharp output image. We demonstrated the effectiveness of the proposed defocus deblurring architecture through the *NTIRE2021 Defocus Deblurring Challenge [1] [4]*.

References

[1] NTIRE 2021 Defocus Deblurring using Dual-pixel Images Challenge. = https://competitions.codalab.org/competitions/28049, 2021. 1, 2, 4, 8

[2] Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S. Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Ian Goodfellow, Andrew Harp, Geoffrey Irving, Michael Isard, Yangqing Jia, Rafal Jozefowicz, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dandelion Mané, Rajat Monga, Sherry Moore, Derek Murray, Chris Olah, Mike Schuster, Jonathon Shlens, Benoit Steiner, Ilya Sutskever, Kunal Talwar, Paul Tucker, Vincent Vanhoucke, Vijay Vasudevan, Fernanda Viégas, Oriol Vinyals, Pete Warden, Martin Wattenberg, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng. TensorFlow: Large-scale machine learning on heterogeneous systems, 2015. Software available from tensorflow.org. 4

[3] Abdullah Abuolaim and Michael S Brown. Defocus deblurring using dual-pixel data. In *Eur. Conf. Comput. Vis.*, pages –. Springer, 2020. 1, 2, 4, 5, 6, 7

[4] Abdullah Abuolaim, Radu Timofte, Michael S Brown, et al. NTIRE 2021 challenge for defocus deblurring using dual-pixel images: Methods and results. In *IEEE Conf. Comput. Vis. Pattern Recog. Worksh.*, 2021. 1, 2, 4, 8

[5] A. Buades, B. Coll, and J. . Morel. A non-local algorithm for image denoising. In *IEEE Conf. Comput. Vis. Pattern Recog.*, volume 2, pages 60–65 vol. 2, 2005. 4

[6] L. D'Andrès, J. Salvador, A. Kochale, and S. Süsstrunk. Non-parametric blur map regression for depth of field extension. *IEEE Trans. Image Process.*, 25(4):1660–1673, 2016. 2

[7] Jun Fu, Jing Liu, Jie Jiang, Yong Li, Yongjun Bao, and Hanqing Lu. Scene segmentation with dual relation-aware attention network. *IEEE Transactions on Neural Networks and Learning Systems*, 2020. 2

[8] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. *Int. Conf. Comput. Vis.*, 1502, 02 2015. 4

[9] A. Karaali and C. R. Jung. Edge-based defocus blur estimation with adaptive scale selection. *IEEE Trans. Image Process.*, 27(3):1126–1137, 2018. 2

[10] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In Yoshua Bengio and Yann LeCun, editors, *Int. Conf. Learn. Represent.*, 2015. 4

[11] A.L. Maas, A.Y. Hannun, and A.Y. Ng. Rectifier nonlinearities improve neural network acoustic models. In *Int. Conf. Machi. Learn.*, Atlanta, Georgia, 2013. 4

[12] Vinod Nair and Geoffrey Hinton. Rectified linear units improve restricted boltzmann machines vinod nair. In *Int. Conf. Machi. Learn.*, volume 27, pages 807–814, 06 2010. 2, 4

[13] J. Park, Y. Tai, D. Cho, and I. S. Kweon. A unified approach of multi-scale deep and hand-crafted features for defocus estimation. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 2760–2769, 2017. 1

[14] J. Shi, L. Xu, and J. Jia. Discriminative blur detection features. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 2965–2972, 2014. 2

[15] Huixuan Tang and Kiriakos N. Kutulakos. Utilizing optical aberrations for extended-depth-of-field panoramas. In *Asean. Conf. Comput. Vis.*, page 365–378, 2012. 1

[16] A. G. Vien, H. Park, and C. Lee. Dual-domain deep convolutional neural networks for image demoireing. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 1934–1942, 2020. 2

[17] X. Wang, R. Girshick, A. Gupta, and K. He. Non-local neural networks. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 7794–7803, 2018. 4

[18] W. Zhao, F. Zhao, D. Wang, and H. Lu. Defocus blur detection via multi-stream bottom-top-bottom network. *IEEE Trans. Pattern Anal. Mach. Intell.*, 42(8):1884–1897, 2020. 2