Efficient generative model for motion deblurring

Han Xiang1,2, Haiwei Sang3, Lilei Sun2, Yong Zhao4

1 School of Information and Engineering, Guangdong Medical University, Xincheng Road, Dongguan, People’s Republic of China
2 College of Computer Science & Technology, Guizhou University, Huaxi Road, Guiyang, People’s Republic of China
3 School of Mathematics and Big Data, Guizhou Education University, Gaoxin Road, Guiyang, People’s Republic of China
4 School of Electronic and Computer Engineering, Peking University Shenzhen Graduate School, Lishui Road, Shenzhen, People’s Republic of China

E-mail: yongzhao@puksz.edu.cn

Abstract: This article proposes a generate model for motion deblurring based on generative adversarial network. The generate model adopts multi-level and multi-scale feature fusion structure. By concatenating different scales of images and feature maps and adding them by pixels, the image details of each level are obtained. The enlargement part of feature maps uses three branches, which can generate more rich and realistic detail. In the training process, three loss functions at the pixel level and the abstract level are used to make the training convergence faster and effectively assist the parameter learning of the generated model. Experiments show that the proposed approach is more efficient than many state-of-the-art methods.

1 Introduction

In the daily process of image capture, image blurring is easy to occur because of the jitter of image capture equipment and the motion of objects. Ideally, blurring in image capturing can be solved by improving illumination conditions, shortening exposure time and reducing shooting distance. However, in the daily environment, there is no such condition, only through the post-processing stage, deblurring as a viable option.

Initial research focused on the non-blind deconvolution method, which assumed that the fuzzy core was known and restored a clear picture in this case, but the fuzzy core was generally unpredictable, which lacked practical value.

Blind deblurring is a commonly used deblurring method. It attempts to estimate the blurred core under unknown conditions and use it to remove the blurring caused by jitter or motion of objects and output clear images.

As of the convolution operation characteristics of blurred images, it is very suitable to use supervised learning method to estimate its convolution core. Therefore, the algorithm of this chapter is to design an effective generating network based on convolutional neural network (CNN) and the idea of blind de-ambiguity to remove all kinds of ambiguity.

2 Related work

Generally, blind deblurring algorithm can be divided into two categories: single image deblurring and multi-image deblurring algorithm.

Multi-image deblurring usually uses multi-angle images or images in time series to remove blur. Although the effect is good, it is not suitable for single image deblurring, which is more common in real environment.

Single image deblurring algorithm is suitable for reality. This kind of method does not need additional hardware requirements and is suitable for more image devices. Fergus et al. [1] proposed a deblurring algorithm based on gradient distribution model. This method focuses on the estimation of the fuzzy kernel. According to the gradient distribution of the blurred image and the non-blurred image, the algorithm constructs a joint posterior probability of the original image and the fuzzy kernel in the case of known observed image. Jia et al. [2] add spatial random model of blurred noise and local smoothing priori knowledge to the probability model, and alternately estimates the restoration process of blurred kernels and clear images until convergence. Cai et al. [3] used frame and curve transform to obtain sparse representation of fuzzy kernels and images. Xu et al. [4] use sparse prior to propose a two-step fuzzy kernel estimation algorithm, which can accurately estimate the fuzzy kernel, after that, the model is used to estimate the clear image.

With the rapid development of deep learning, convolution neural network utilises massive image data and very depth model to learn the feature expression of image at multiple levels and has achieved remarkable results in the field of computer vision, many deep learning-based blind deblurring approaches have been proposed. Yan et al. [5] proposed the algorithm consists of a deep neural network and a general regression neural network making full use of both the classification ability of DNN and the regression ability of GRNN. Sun et al. [6] constructed a model to reconstruct the overall clear image after predicting multiple local blurred information of the image through CNN. Hradis et al. [7] first use CNN to generate a set of blurred image data and then use clear image and blurred image as training data pairs to train a blind deblurring network for text image blurring removal. Tao et al.[8] used Resnet [9] modules and LSTM [10] structure to construct CNN model, and achieved satisfactory deblurring effect, especially in the image details, the recovery is quite clear.

Generative Adversarial Network (GAN) [11] proposed by Ian Goodfellow et al. is a hot area of deep learning. It contains two trainable models: generating model G, discriminating model D. G and D are constantly confronting the game. Ideally, the two will eventually achieve a dynamic balance. The images generated by G are infinitely close to the real distribution data, while D cannot distinguish the authenticity of the results generated by G. Baseon the ideas of conditional GAN [12] and perceptual loss [13], Kupyn et al. [14] proposed a deblurring algorithm for the first time and surpasses many state-of-the-art methods.

Ideas of CNN, GAN and multi-level and multi-scale feature fusion structure are adapted to design the generate model in this paper.

3 Methodology

We designed our two models: generate model G and discriminant model D based on previous researches on GAN.
The discriminant model is a typical classification network, which consists of six down-sampling modules and one output function layer. Each sub-sampling module includes convolution layer, batch normalisation and PReLU [15] activation function layer. The final output function uses sigmoid, 0 represents false and 1 represents true. Different from the ordinary classification network, the discriminant network removes the pooling layer and directly uses the convolution layer with a stride of 2, thus reducing the computational complexity. Each down-sampling layer reduces the length and width of the input layer to 1/2, respectively. There are six down-sampling layers. The original image can be reduced to 1/64. The size of the input image is 256 × 256, so the result of down-sampling is 4 × 4. Finally, the network output channel is 1 by using 4 × 4 convolution core to connect the sigmoid function Fig. 1.

3.2 Generate model

Our focus on designing generates model G which generates more output results that is close to a clear image. FCN can generate output results of the same size as the original input graph, so the idea of FCN algorithm is also used to construct G. As the generator of G requires higher precision than the ordinary segmentation network, it cannot directly use the existing FCN network. The G network of this chapter uses the idea of Unet to design symmetrical Encoder and Decoder, which reduces the computational complexity. Each down-sampling layer reduces the length and width of the input layer to 1/2, respectively. There are two white rectangular boxes are Resblock layers [9].

The proposed discriminant model

Fig. 1 The proposed discriminant model

Fig. 2 The proposed generate model

3.1 Discriminant model

The discriminant model is a typical classification network, which consists of six down-sampling modules and one output function layer. Each sub-sampling module includes convolution layer, batch normalisation and PReLU [15] activation function layer. The final output function uses sigmoid, 0 represents false and 1 represents true. Different from the ordinary classification network, the discriminant network removes the pooling layer and directly uses the convolution layer with a stride of 2, thus reducing the computational complexity. Each down-sampling layer reduces the length and width of the feature graph can be reduced to 1/2 by using bilinear, because interpolation smoothing operation will lose a lot of details, which is not suitable for such generation tasks. The algorithm in this chapter designs a composite up-sampling structure, which is composed of a specified magnification calculation.

In addition to bilinear interpolation, CNN has many kinds of up-sample operations, unpooling [16] is one of them, as shown in the following figure: Fig. 3.

Pooling operation down sample the feature maps. For example, when Max pooling is used, the position of each convolution core covering the maximum value is recorded, and the position of the maximum value is restored by using this value in the up-sample operation.

Transposed convolution is inserting a value of 0 into the original graph and following a convolution, as shown in the following figure: Fig. 4.

Main idea of Pixel shuffle is to convolute the original feature maps with shape \((r \times W, r \times H)\) to \(c \times r^2\) channels, where \(c\) is the number of channels, \(W\) and \(H\) are the width and height of the feature maps, \(r\) is the scale factor, then reduce the feature maps of every \(r^2\) channel to one channel with shape \((r \times W, r \times H)\). It is
equivalent to combining the pixels in \( r' \) feature graphs into \( r' \) pixels in one enlarged image, which is realised by using the following formula:

\[
PS(T)_{3,3,3} = T[\cdot] : C \times \text{mod}(r', r) + C \times \text{mod}(r, r') + C
\]

(1)

We also adopt Pixel shuffle as a branch of up-sampling unit. Above-mentioned three branches are used to up-sample the feature map to the required scale. At the end of the three branches, the enlarged feature map is added by pixels.

### 3.3 Losses

We use the cross-entropy loss function with a gradient penalty for discriminant model to judge the output of the two classifications as mentioned in [14]. In the following formula, \( L_{\text{blur}} \), \( L_{\text{sharp}} \), \( L_d \) stand for the blur image, sharp image and computed loss for discriminant model, respectively.

\[
L_d = D(G(\text{blur})) - D(G(\text{sharp})) + \text{penalty}
\]

(2)

Generally, the difference between the generated image and the clear image label can be expressed by \( \text{MSE} \) [18], which compares the mean square deviation of each pixel. If the blurred image is caused by large-scale jitter, the distance between the corresponding pixels of the blurred and clear image will be large, and the loss calculated from it will be very large, and the same large gradient will be calculated in the process of BP, which will cause the adjustment oscillation of the weights of each neuron and make it difficult to converge. We use SmoothL1 loss [19] to calculate the residual between the generated image and the clear image, it can be formulated as follows:

\[
L_{\text{smooth}}(\cdot) = \begin{cases} 
0.5x^2 & \text{if } |x| < 1 \\
|x| - 0.5 & \text{otherwise}
\end{cases}
\]

(3)

when the absolute value is <1, the loss is calculated by using Euclidean distance, and in other cases, 0.5 is subtracted, thus avoiding the problem that the value is too large to converge after derivation.

We adopt perceptual losses [13] as another loss. Perceptual losses compares two images in feature space and uses MSE-like formula to express the difference. Experiments show that our approach has a good perception loss.

\[
L_{\text{feat}}(\hat{y}, \tilde{y}) = \frac{1}{C_iH_iW_i} \| \varphi_i(\hat{y}) - \varphi_i(\tilde{y}) \|^2
\]

(4)

where \( C_i \) is output number of convolution layer, \( H_i \) and \( W_j \) are height and width, respectively, \( \varphi_i \) is CNN activation function and \( \hat{y} \) and \( \tilde{y} \) are the real and generated image, respectively. We use VGG16 structure as feature extraction network and use Layer 16 to calculate perceptual loss.

The final loss is as follows:

\[
L = L_d + \lambda_1 \times L_{\text{smooth}} + \lambda_2 L_{\text{feat}}
\]

(5)

Here \( \lambda_1 \) is coefficient of SmoothL1 loss with a value of 10, \( \lambda_2 \) is that for perceptual losses, with a value of 100 as the same settings in DeblurGan [14], \( L \) stands for the total computed loss of mentioned model. \( L_d \), \( L_{\text{smooth}} \), \( L_{\text{feat}} \) stand for the contents mentioned in Formula (2)–(4), respectively.
References

[1] Fergus, R., Singh, B., Hertzmann, A., et al.: ‘Removing camera shake from a single photograph’. Proc. Int. Conf. Computer Graphics and Interactive Techniques, Kuala Lumpur, Malaysia, November 2006, vol. 25, (3), pp. 787–794

[2] Shan, Q., Jia, J., Agarwala, A.: ‘High-quality motion deblurring from a single image’, ACM Trans. Graph., 2008, 27, (3), pp. 1–10

[3] Cai, J.F., Ji, H., Liu, C., et al.: ‘Blind motion deblurring from a single image using sparse approximation’. Proc. Int. Conf. Computer Vision and Pattern Recognition, Miami, USA, June 2009, pp. 104–111

[4] Xu, L., Zheng, S., Ju, J.: ‘Unnatural 0 sparse representation for natural image deblurring’. Proc. Int. Conf. Computer Vision and Pattern Recognition, Oregon, Portland, June 2013, pp. 1107–1114

[5] Yan, R., Shao, L.: ‘Blind estimation of blur kernels and parameters from a single image’, IEEE Trans. Image Process. A, Signal Process. Soc., 2016, 25, (4), pp. 1–11

[6] Jian, S., Cao, W., Xu, Z., et al.: ‘Learning a convolutional neural network for non-uniform motion blur removal’. Proc. Int. Conf. Computer Vision and Pattern Recognition, Boston, USA, June 2015, pp. 769–777

[7] Hradis, M., Kotera, J., Zemecik, P., et al.: ‘Convolutional neural networks for direct text deblurring’. Proc. Int. Conf. British Machine Vision, Swansea, Britain, May 2015, vol. 10, p. 2

[8] Tao, X., Gao, H., Shen, X., et al.: ‘Scale-recursive network for deep image deblurring’. Proc. Int. Conf. Computer Vision and Pattern Recognition, Salt Lake City, USA, June 2018, pp. 8174–8182

[9] He, K., Zhang, X., Ren, S., et al.: ‘Deep residual learning for image recognition’. Proc. Int. Conf. Computer Vision and Pattern Recognition, LAVegas, USA, June 2016, pp. 770–778

[10] Xingjian, S., Chen, Z., Wang, H., et al.: ‘Convolutional LSTM network: A machine learning approach for precipitation nowcasting’. Proc. Int. Advances in neural information processing systems, Montreal, Canada, December 2015, pp. 802–810

[11] Goodfellow, I.J., Pougetbadie, J., Mirza, M., et al.: ‘Generative adversarial nets’. Proc. Int. Neural Information Processing Systems, Montreal, Canada, December 2014, pp. 2672–2680

[12] Mirza, M., Osindero, S.: ‘Conditional generative adversarial nets’. arXiv preprint arXiv:1411.1784, 2014

[13] Johnson, J., Alahi, A., Feifei, L.: ‘Perceptual losses for real-time style transfer and super-resolution’. Proc. Int. European Conf. on Computer Vision, Amsterdam, The Netherlands, October 2016, pp. 694–711

[14] Kupyn, O., Budzan, V., Mykhailych, Y., et al.: ‘Deblurgan: blind motion deblurring using conditional adversarial networks’. Proc. Int. Computer Vision and Pattern Recognition, Salt Lake City, USA, June 2018, pp. 8183–8192

[15] He, K., Zhang, X., Ren, S., et al.: ‘Delving deep into rectifiers: surpassing human-level performance on ImageNet classification’. Proc. Int. Computer Vision and Pattern Recognition, Boston, USA, June 2015, pp. 1026–1034

[16] Noh, H., Hong, S., Han, B.: ‘Learning deconvolution network for semantic segmentation’. Proc. Int. Computer Vision and Pattern Recognition, Boston, USA, June 2015, pp. 1520–1528

[17] Shi, W., Caballero, J., Huszar, F., et al.: ‘Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network’. Proc. Int. Computer Vision and Pattern Recognition, LAVegas, USA, June 2016, pp. 1874–1883

[18] Wang, Z., Bovik, A.C.: ‘A universal image quality index’, IEEE Signal Process. Lett., 2002, 9, (3), pp. 81–84

[19] Ren, S., He, K., Girshick, R., et al.: ‘Faster R-CNN: towards real-time object detection with region proposal networks’, IEEE Trans. Pattern Anal. Mach. Intell., 2017, 39, (6), pp. 1137–1149

[20] Nah, S., Hyun Kim, T., Mu Lee, K.: ‘Deep multi-scale convolutional neural network for dynamic scene deblurring’. Proc. Int. European Conf. on Computer Vision, Munich, Germany, November 2012, pp. 27–40

Fig. 6 Result comparisons on testing dataset (a) DeblurGan, (b) Ours, (c) Sharp