TWO-STREAM MULTI-FOCUS IMAGE FUSION BASED ON THE LATENT DECISION MAP

*Weihong Zeng, Fei Li, Hongyu Huang, Yue Huang, Xinghao Ding**

Fujian Key Laboratory of Sensing and Computing for Smart City,
School of Information Science and Engineering, Xiamen University, China
*dxh@xmu.edu.cn

ABSTRACT

The multi-focus image fusion with deep learning methods is mostly regarded as a two or three-category problem. Current systems utilize sliding windows to classify each pixel into focused or defocused, which is time consuming and requires post-processing such as denoising. In this paper, we propose a novel network architecture for multi-focus image fusion based on the latent decision map. For a regression task instead of a classification problem, we focus on learning the latent spatial decision map. This decision map indicates the degree of each focused pixel. To further improve the fusion result, we utilize the ResNet blocks to extract image features, and then combine low-level features with high-level semantic information. Our apporach makes the learning process easier and has better robustness and efficiency as well. Experimental results demonstrate that our framework has ability of achieving the state-of-the-art in terms of both qualitative and quantitative measures.

***Index Terms*—** Multi-Focus, Image Fusion, Two-Stream Feature Extraction, Latent Decision Map

1. INTRODUCTION

Multi-Focus Image Fusion (MFIF) aims to reconstruct a clear, fully focused image by fusing multi-focus images with the same scene. Multi-focus image fusion technology has a wide range of applications in civil digital, remote sensing, biomedical research and other fields. There are some traditional methods, including boundary finding (BF) [1], guided filtering (GF) [2], image matting (IM) [3] and dense SIFT (DSIFT) [4]. Recently, Liu et al.[5] applied deep convolutional neural networks to multi-focus image fusion firstly. They treated it as a two-category issue and classified per pixel to "focused" and "unfocused". Tang et al.[6] improved this method and repurposed the problem to "focused", "unfocused" and "difficult to judge" and resulted in better quality metrics. These two methods regarded the multi-focus image fusion problem as a simple classification task and utilized a series of post-processing methods to remove the noise. Their solution is complicated and non-end-to-end. Yan et al.[7] proposed an unsupervised way to reconstruct clear images. The multi-focus image dataset was directly used as the training set, and the distribution of the dataset is consistent with the test set. This approach achieved better results for the mentioned reasons. Yet, it is challenged to collect so many focus images for training in practical application. The proposed methods can not be applied to other databases and must be trained individually and repeatly for each database.

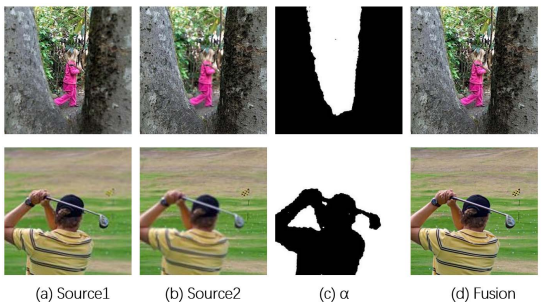

Fig. 1. Our method for multi-focus image fusion. (a) and (b) are two input images of different focus, (c) is the decision map. (d) is the final fusion result.

In order to solve the issue, our method introduces domain knowledge, generating an all-focused image through our end-to-end network. It is relatively difficult to reconstruct images directly, so we introduce the decision map α to express the depth of the field. Our network can fuse the images at pixel level with better edge and small objects of different depths-of-field. Our contributions can be summarized as follows,

- Our network generates fully focused image from latent spatial decision map α, which is an end-to-end system. Compared with other methods, our network does not need post-processing and classifies small objects more precisely.
- Inspired by domain knowledge, we introduce an latent spatial decision map α to learn the degree of the fo-

This work was supported in part by the National Natural Science Foundation of China under Grants 61571382, 81671766, 61571005, 81671674, 61671309 and U1605252, in part by the Fundamental Research Funds for the Central Universities under Grant 20720160075, 20720180059 in part by the CCF-Tencent open fund, and the Natural Science Foundation of Fujian Province of China(No.2017J01126)

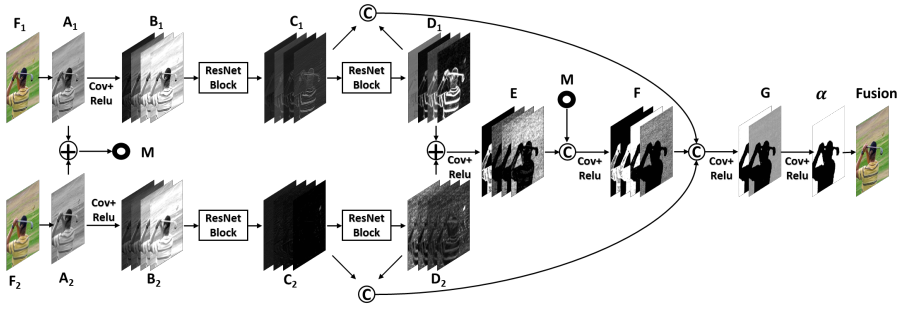

Fig. 2. Our model review. "+" indicates two matrices addition; "C" in the circle indicates that the matrices are concatenated in the third channel. To simplify the figure, the bold ring represents that we concat the matrix M with the feature maps E.

cus at each pixel. We consider this task as a regression problem rather than the two-category or three-category problem, which empowers the transition of the depth in the field.

- We boost the feature fusion by using two-stream feature extraction. Both low-level features and high-level features are merged, which avoids gradient vanishing and makes learning process easier.

- We don't need real multi-focus images to train. Though the images we trained are different from the test datasets, we have superior generalization performance on all multi-focus datasets than other methods.

2. METHOD

We illustrate our network in Fig. 2. Overall, our network can generate fully-focused clear images with different focus and any size.

2.1. Deep Detail Network for Image Fusion

2.1.1. Two-Stream Feature Extraction Part

As shown in Fig. 2, we use ResNet blocks [8]to extract network features. Gray images are fed into our network, which can also simultaneously apply to RGB and gray multi-focus images. We extract the features separately in each stream to obtain the different information through the different focus input. Feature extraction with two ResNet blocks is available to get more high-level semantic features. Moreover, the difference features of one stream and other stream can focus on respective clear part and make up to each other to get a decision map. This part can be expressed as follows,

$$\begin{aligned} Y^1 &= max(W^1 * A_1 + b^1, 0), \\ Y^{2L} &= max(W^{2L} * Y^{2L-1} + b^{2L}, 0), \\ Y^{2L+1} &= max(W^{2L+1} * Y^{2L} + b^{2L+1}, 0) + Y^{2L-1}, \end{aligned} \tag{1}$$

where W contains weights, b is biases and $L = 1, 2$. A_1 stands for the input (gray images). When $L = 1$, $Y^{2*1+1} = C_1$ and when $L = 2$, $Y^{2*2+1} = D_1$. A_1, C_1, D_1 can be seen in Fig. 2. The feature extraction of second stream is the same with first one, except input image A_2.

2.1.2. Image Feature Fusion Part

In order to fuse two-stream features together, we add the D_1 and D_2 generated from two-stream. Though D_1 and D_2 contain much more high-level features, the lack of details in the boundary regions is an enormous challenge for reconstruction of decision map. Therefore, we concatenate the original features M to E, where M is the addition of A_1 and A_2. It is of great importance to make the F contain more detail features. The concat function is defined as stack of feature maps in the third dimension. To make decision map contain the two-stream features, the output F and the layer-by-layer features C_1,C_2,D_1 and D_2 are concatenated. In other words, the decision map can be refined by embedding more details from two-stream extraction part. This skip connection can also directly propagate loss throughout the entire network, which is useful for estimating the decision map α and the final fusion image. Image feature fusion part can be expressed as follows,

$$\begin{aligned} E &= max(W^E * (D_1 + D_2) + b^E, 0), \\ F &= max(W^F * Concat(M, E) + b^F, 0), \\ G &= max(W^G * Concat(C_1, D_1, F, C_2, D_2) + b^G, 0), \end{aligned} \tag{2}$$

where W contains weights and b biases. E, F, G, C_1, D_1, C_2, D_2 can be seen in Fig.2.

2.1.3. Multi-Focus Image Reconstruction part

Since We have acquired enough low-level and high-level features based on the image fusion process, we reduce the number of output feature maps to 64 dimensions through a convolutional layer and a rectified linear unit (Relu)[9] and reduce it to 1 dimension finally. That is the decision map α, leading to the final regression problem much easier. Finally, two-

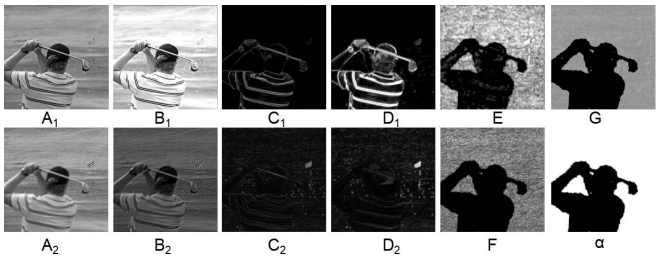

Fig. 3. Visualization of the network-learning decision map process. The naming of all the images corresponds to the names in Fig. 2. Each of them is a feature map for each stage.

dimensional spatial decision map with the input image size is obtained, then we can actively learn the degree of image focus at the pixel level. The final full-clear image is generated by the relationship between the decision map and the input image as follows,

$$Fusion = \alpha \cdot F_1 + (1 - \alpha) \cdot F_2, \tag{3}$$

where α is the decision map, F_1 and F_2 represent the two input focus images.

2.2. Loss Function

Our network is based on learning of the decision map. The quality of the decision map has a huge impact on the quality of the reconstructed image. L_2 loss make reconstructed image more smooth[10], so we use the L_1 loss[11] as final loss function. Our goal is to learn the appropriate decision map α, so the reconstructed image is the fully focused image. The loss function is defined as,

$$Loss(F_1, F_2, \alpha) = \parallel \alpha \cdot F_1 + (1 - \alpha) \cdot F_2 - GT \parallel_1, \tag{4}$$

where GT is GroundTrue of clear image. In the training stage, α learns a continuous value between 0 and 1. But in the test stage, we will generate the decision map based on the degree of the focus, and finally we obtain binary decision map containing only 0 and 1. For the decision map, we define α as follows,

$$\alpha = \begin{cases} 1, & \alpha(x, y) \geq 0.5 \\ 0, & \alpha(x, y) < 0.5, \end{cases} \tag{5}$$

where (x, y) is the coordinates of the image. It's worth mentioning that we don't need to do this during the training stage. It's just a layer added in the test stage.

To understand how our network works, we visualize the network and take a feature map after each layer to view the learning process of the decision map. It can be seen from the Fig. 3. The top and bottom of stream extract and learn the focused portion of the corresponding multi-focus images in the two-stream structure respectively. Taking the "GOLF" image as an example, after the top and bottom of stream are extracted by the feature map, the upper stream focus on the player, while the nether stream focuses on flag, the golf ball scattered on the ground and the background. In the part of fusion, foreground and background area begins to be segmented. At the same time, the noise particles become smaller and smaller and numerical value of α approaches 0 or 1. The final decision map can combine the respective focus information to fuse the focus area and generate final clear image.

3. EXPERIMENT

3.1. Training Set and Network Parameter Settings

We use high-resolution images from the DIV2K train set [12] to simulate multi-focus images as training sets. We randomly crop 60000 images with the size of 32×32. Six templates are learnt from PCNN [6] as shown in Fig.4. We add gaussian blur [13] to the original image with a standard deviation 1 and filter sizes of 3, 7, 11 and 15 to simulate the different degree of focus respectively, which produces a total of $60000 \times 6 \times 4 = 1.44$ million pair of images. During training, we randomly exchange training images for the position of two-stream, which makes the network to understand whether its learning goal is to focus or not, rather than other tasks and make network network have better generalization capabilities.

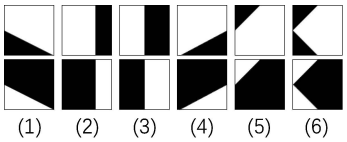

Fig. 4. six templates for gaussian blur. The black part uses guassian blur by pixel, and the white part keeps the value unchanged.

All the layer's feature maps are 128, kernel size is 3×3 and step size is 1, except G is 64 and α is 1 shown in Fig.2. The Adam [14] optimization method was used to train network for a total of 200,000 training iterations. The batch size is set to 32. The initial learning rate is set to 0.0001 and dividing it by 10 at 5×10^4 and 1.5×10^5 iterations. All experiments were implemented with Tensorflow [15] on Nvidia 1080TI GPU.

3.2. Compared with Other Methods

We compare ours method with several of the best methods on multi-focus image datasets. Including CNN [5], PCNN [6], BF [1], GF [2], IM [3] and DSIFT [4]. We validated various algorithms on two published multi-focus image datasets which is widely used to evaluate fusion performances for a total of 39 pairs of multi-focus images. 19 pairs of them are

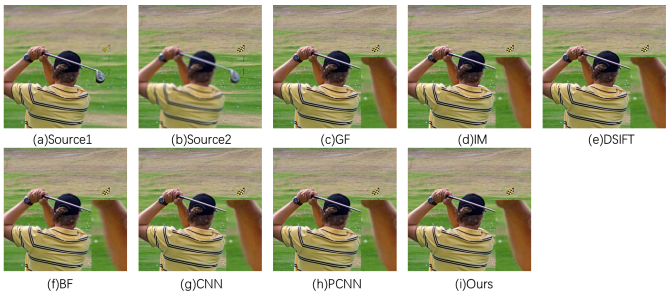

Fig. 5. Fusion results of various methods on "GOLF" images.

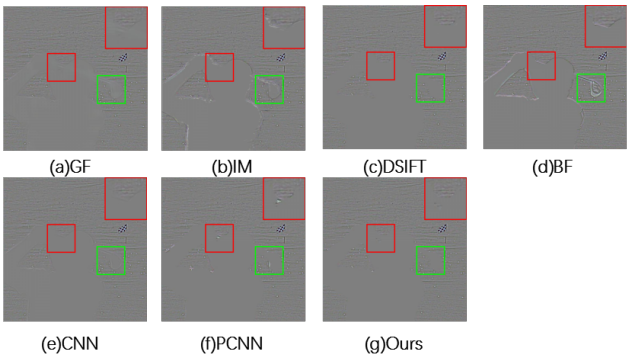

Fig. 6. The residual of difference images between each fused image and the first source image (Source1).

gray images from the multi-focus image fusion dataset. The other 20 pairs are RGB image from "Lytro" dataset [16]. No single quality metric can fully explain the quality of the fused image, so it is necessary to combine multiple quality metric to evaluate the the fused images. We select four quality metric to evaluate the fused images, The first one is normalized mutual information Q_{MI} [17]. The second one is Objective Pixel-level Image Fusion Performance Measure Q_G [18]. The third one is information entropy EN [19], which estimates the amount of information present fusion image. The last one is visual information fidelity $VIFF$ [20], which measures the visual information fidelity.

In Fig. 5, we compare the fusion results of our proposed network and other methods for the "GOLF" image in the "Lytro" image dataset. It can be seen that our method generates the best fusion result, which have better visual effect. We select the partial magnification to see the arm area of the fused image. It can be seen that in the edge part of the left arm, IM, BF, CNN, have obviously edge blur and our fusion image is the clearest. By comparing the resulting fused image with the source image, we can see that ours method of the transition between the two focous regions is more natural.

In Fig. 6, It can be clearly seen that the IM, BF method have unsatisfied performances on the edge and have so much noise. IM and BF methods are less effective at the shape of edge and the bottom of golf club head is severely deformation.

GOLF	$Q_{MI}\uparrow$	$Q_G\uparrow$	$EN\uparrow$	$VIFF\uparrow$
GF	1.050	0.720	7.011	0.966
IM	1.073	0.710	**7.018**	0.966
DSIFT	1.145	0.725	7.013	0.968
BF	1.123	0.715	7.004	0.946
CNN	1.089	0.720	7.007	0.962
PCNN	1.143	0.723	7.013	0.964
Ours	**1.149**	**0.727**	7.014	**0.969**

Table 1. Comparison of performance among our method and other methods on "GOLF" image fusion.

PCNN has more noise at the bottom of the golf club head, left shoulder and left arm. Though GF and CNN extracts most detail in source images, but the merging effect is also worse than our method in boundary regions. DSIFT is the best non-deep learning method for fusion. But it worth paying attention to that DSIFT and other methods have lost the tiny triangular regions between the right hand and the hat. As visualized in the red box we circled, it should be background but they misclassification. This tiny triangular regions was reserved well in our method, because we never using hold filling [5, 6] to remove noise. Our method preserve the smooth area in foreground and background well and the transition on the edge details more natural and has a better visual perception.

Dataset	$Q_{MI}\uparrow$	$Q_G\uparrow$	$EN\uparrow$	$VIFF\uparrow$
GF	1.073	0.705	7.396	0.884
IM	1.146	0.707	7.393	0.879
DSIFT	1.190	0.713	7.389	0.885
BF	1.189	**0.716**	7.384	0.876
CNN	1.154	0.714	7.389	0.884
PCNN	1.189	0.709	7.387	0.879
Ours	**1.192**	0.712	**7.401**	**0.888**

Table 2. Comparison of performance among our method and other methods on all the 39 images in both of datasets.

Table. 2 lists the fusion quality metrics calculated using the above four metrics and the average of the fusion results produced by 39 pairs of multi-focus images. The best result is bolded on each column. The above metrics show that our method has reached the best performance and significant improvement over other methods visually and metrically.

4. CONCLUSION

In this paper, we propose an end-to-end multi-focus image fusion framework based on the latent decision map. Our network is capable of extracting most important feature and incorporating low-level features with high-level features through the two-stream. We adopt the simple and efficient L_1 loss and learn the appropriate decision map from two-stream to reconstruct full-focused image finally. Our method does not need to use real multi-focus images for training and post-processing such as denoising. Quantitative and qualitative results demonstrate that our contributions lead our network to achieve the state-of-the-art.

5. REFERENCES

[1] Yu Zhang, Xiangzhi Bai, and Tao Wang. Boundary finding based multi-focus image fusion through multi-scale morphological focus-measure. *Information fusion*, 35:81–101, 2017.

[2] Shutao Li, Xudong Kang, and Jianwen Hu. Image fusion with guided filtering. *IEEE Transactions on Image Processing*, 22(7):2864–2875, 2013.

[3] Shutao Li, Xudong Kang, Jianwen Hu, and Bin Yang. Image matting for fusion of multi-focus images in dynamic scenes. *Information Fusion*, 14(2):147–162, 2013.

[4] Yu Liu, Shuping Liu, and Zengfu Wang. Multi-focus image fusion with dense sift. *Information Fusion*, 23:139–155, 2015.

[5] Yu Liu, Xun Chen, Hu Peng, and Zengfu Wang. Multi-focus image fusion with a deep convolutional neural network. *Information Fusion*, 36:191–207, 2017.

[6] Han Tang, Bin Xiao, Weisheng Li, and Guoyin Wang. Pixel convolutional neural network for multi-focus image fusion. *Information Sciences*, 433:125–141, 2018.

[7] Xiang Yan, Syed Zulqarnain Gilani, Hanlin Qin, and Ajmal Mian. Unsupervised deep multi-focus image fusion. *arXiv preprint arXiv:1806.07272*, 2018.

[8] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[9] George E Dahl, Tara N Sainath, and Geoffrey E Hinton. Improving deep neural networks for lvcsr using rectified linear units and dropout. In *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*, pages 8609–8613. IEEE, 2013.

[10] Wei-Sheng Lai, Jia-Bin Huang, Narendra Ahuja, and Ming-Hsuan Yang. Deep laplacian pyramid networks for fast and accurate superresolution. In *IEEE Conference on Computer Vision and Pattern Recognition*, volume 2, page 5, 2017.

[11] Hang Zhao, Orazio Gallo, Iuri Frosio, and Jan Kautz. Loss functions for image restoration with neural networks. *IEEE Transactions on Computational Imaging*, 3(1):47–57, 2017.

[12] Eirikur Agustsson and Radu Timofte. Ntire 2017 challenge on single image super-resolution: Dataset and study. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, July 2017.

[13] Robert A Hummel, B Kimia, and Steven W Zucker. Deblurring gaussian blur. *Computer Vision, Graphics, and Image Processing*, 38(1):66–80, 1987.

[14] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[15] Martín Abadi, Paul Barham, Jianmin Chen, Zhifeng Chen, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Geoffrey Irving, Michael Isard, et al. Tensorflow: a system for large-scale machine learning. In *OSDI*, volume 16, pages 265–283, 2016.

[16] http://mansournejati.ece.iut.ac.ir/content/lytro-multi-focus-dataset.

[17] M Hossny, S Nahavandi, and D Creighton. Comments on'information measure for performance of image fusion'. *Electronics letters*, 44(18):1066–1067, 2008.

[18] CS Xydeas, , and Vladimir Petrovic. Objective image fusion performance measure. *Electronics letters*, 36(4):308–309, 2000.

[19] BK Shreyamsha Kumar. Image fusion based on pixel significance using cross bilateral filter. *Signal, image and video processing*, 9(5):1193–1204, 2015.

[20] Yu Han, Yunze Cai, Yin Cao, and Xiaoming Xu. A new image fusion performance metric based on visual information fidelity. *Information fusion*, 14(2):127–135, 2013.