Information Fusion 36 (2017) 191–207

Contents lists available at ScienceDirect

Information Fusion

journal homepage: www.elsevier.com/locate/inffus

Multi-focus image fusion with a deep convolutional neural network

Yu Liu[a], Xun Chen[a,*], Hu Peng[a], Zengfu Wang[b]

[a] Department of Biomedical Engineering, Hefei University of Technology, Hefei 230009, China
[b] Department of Automation, University of Science and Technology of China, Hefei 230026, China

ARTICLE INFO

Article history:
Received 21 March 2016
Revised 29 November 2016
Accepted 4 December 2016
Available online 5 December 2016

Keywords:
Image fusion
Multi-focus image fusion
Deep learning
Convolutional neural networks
Activity level measurement
Fusion rule

ABSTRACT

As is well known, activity level measurement and fusion rule are two crucial factors in image fusion. For most existing fusion methods, either in spatial domain or in a transform domain like wavelet, the activity level measurement is essentially implemented by designing local filters to extract high-frequency details, and the calculated clarity information of different source images are then compared using some elaborately designed rules to obtain a clarity/focus map. Consequently, the focus map contains the integrated clarity information, which is of great significance to various image fusion issues, such as multi-focus image fusion, multi-modal image fusion, etc. However, in order to achieve a satisfactory fusion performance, these two tasks are usually difficult to finish. In this study, we address this problem with a deep learning approach, aiming to learn a direct mapping between source images and focus map. To this end, a deep convolutional neural network (CNN) trained by high-quality image patches and their blurred versions is adopted to encode the mapping. The main novelty of this idea is that the activity level measurement and fusion rule can be jointly generated through learning a CNN model, which overcomes the difficulty faced by the existing fusion methods. Based on the above idea, a new multi-focus image fusion method is primarily proposed in this paper. Experimental results demonstrate that the proposed method can obtain state-of-the-art fusion performance in terms of both visual quality and objective assessment. The computational speed of the proposed method using parallel computing is fast enough for practical usage. The potential of the learned CNN model for some other-type image fusion issues is also briefly exhibited in the experiments.

1. Introduction

In the field of digital photography, it is often difficult for an imaging device like a digital single-lens reflex camera to take an image in which all the objects are captured in focus. Typically, under a certain focal setting of optical lens, only the objects within the depth-of-field (DOF) have sharp appearance in the photograph while other objects are likely to be blurred. A popular technique to obtain an all-in-focus image is fusing multiple images of the same scene taken with different focal settings, which is known as multi-focus image fusion. At the same time, multi-focus image fusion is also an important subfield of image fusion. With or without modification, many algorithms for merging multi-focus images can also be employed for other image fusion tasks such as visible-infrared image fusion and multi-modal medical image fusion (and vice versa). From this point of view, the meaning of studying multi-focus image fusion is twofold, which makes it an active topic in image processing community. In recent years, various image fusion methods have been proposed, and these methods can be roughly classified into two categories [1]: transform domain methods and spatial domain methods.

The most classic transform domain fusion methods are based on multi-scale transform (MST) theories, which have been applied in image fusion for more than thirty years since the Laplacian pyramid (LP)-based fusion method [2] was proposed. Since then, a large number of multi-scale transform based image fusion methods have appeared in this field. Some representative examples include the morphological pyramid (MP)-based method [3], the discrete wavelet transform (DWT)-base method [4], the dual-tree complex wavelet transform (DTCWT)-based method [5], and the non-subsampled contourlet transform (NSCT)-based method [6]. These MST-based methods share a universal three-step framework, namely, decomposition, fusion and reconstruction [7]. The basic assumption of MST-based methods is that the activity level of source images can be measured by the decomposed coefficients in a selected transform domain. Apart from the selection of MST domain,

* Corresponding author.
E-mail addresses: yuliu@hfut.edu.cn, liuyu1@mail.ustc.edu.cn (Y. Liu), xun.chen@hfut.edu.cn (X. Chen).

http://dx.doi.org/10.1016/j.inffus.2016.12.001

the rules designed for merging decomposed coefficients also play a very important role in MST-based methods, and many studies have also been taken in this direction [8–11]. In recent years, a new kind of transform domain fusion methods [12–16] has emerged as an attractive branch in this field. Different from the above introduced MST-based methods, these methods transform images into a single-scale feature domain with some advanced signal representation theories such as independent component analysis (ICA) and sparse representation (SR). This category of methods usually employs the sliding window technique to pursue an approximate shift-invariant fusion process. The key issue of these methods is to explore an effective feature domain for the calculation of activity level. For instance, as one of the most representative approaches belonging to this category, the SR-based method [13] transforms the source image patches into sparse domain and applies the L1-norm of sparse coefficients as the activity level measurement.

The spatial domain methods in the early stage usually adopt a block-based fusion strategy, in which the source images are decomposed into blocks and each pair of block is fused with a designed activity level measurement like spatial frequency and sum-modified-Laplacian [17]. Clearly, the block size has a great impact on the quality of fusion results. Since the earliest block-based methods [18,19] using manually fixed size appeared, many improved versions have been proposed on this topic, such as the adaptive block based method [20] using differential evolution algorithm to obtain a fixed optimal block size, and some recently introduced quad-tree based methods [21,22] in which the images can be adaptively divided into blocks with different sizes according to image content. Another type of spatial domain methods [23,24] is based on image segmentation by sharing the similar idea of block-based methods, but the fusion quality of these methods relies heavily on the segmentation accuracy. In the past few years, some novel pixel-based spatial domain methods [25–31] based on gradient information have been proposed, which can currently obtain state-of-the-art results in multi-focus image fusion. To further improve the fusion quality, these methods usually apply relatively complex fusion schemes (can be regarded as *rules* in a broad sense) to their calculation results of activity level measurement.

It is well known that for either transform domain or spatial domain image fusion methods, activity level measurement and fusion rule are two crucial factors. In most existing image fusion methods, these two issues are considered separately and designed manually [32]. To make further improvements, many recently proposed methods tend to be more and more complicated on these two issues. In the MST-based methods, new transform domains in [33,34] and new fusion rules in [9–11] were introduced. In the SR-based methods, there were new sparse models and more complex fusion rules in [35–37]. In the block-based methods, new focus measures were proposed in [21,22]. In the pixel-based methods, new activity level measurements were introduced in [27,29] and the fusion schemes employed in [26,28–30] are very intricate. The above introduced works were all published within the last five years. It is worthwhile to clarify that we don't mean these elaborately designed activity level measurements and fusion rules are not important contributions, but the problem is that manual design is really not an easy task. Moreover, from a certain point of view, it is almost impossible to come up with an ideal design that takes all the necessary factors into account.

In this paper, we address this problem with a deep learning approach, aiming to learn a direct mapping between source images and focus map. The focus map here indicates a pixel-level map which contains the clarity information after comparing the activity level measure of source images. To achieve this target, a deep convolutional neural network (CNN) [38] trained by high-quality image patches and their blurred versions is adopted to encode the mapping. The main novelty of this idea is that the activity level measurement and fusion rule can be jointly generated through learning a CNN model, which overcomes the above difficulty faced by existing fusion methods. Based on this idea, we propose a new multi-focus image fusion method in spatial domain. We demonstrate that the focus map obtained from the convolutional network is reliable that very simple consistency verification techniques can lead to high-quality fusion results. The computational speed of the proposed method using parallel computing is fast enough for practical usage. At last, we briefly exhibit the potential of the learned CNN model for some other-type image fusion issues, such as visible-infrared image fusion, medical image fusion and multi-exposure image fusion.

To the best of our knowledge, this is the first time that the convolutional neural network is applied to an image fusion task. The most similar work was proposed by Li et al. [19], in which they pointed out that the multi-focus image fusion can be viewed as a classification problem and presented a fusion method based on artificial neural networks. However, there exist significant differences between the method in [19] and our method. The method in [19] first calculates three commonly used focus measures (feature extraction) and then feeds them to a three-layer (input-hidden-output) network, so the network just acts as a classifier for the fusion rule design. As a result, the source images must be fused patch by patch in [19]. In this work, the CNN model is simultaneously used for activity level measure (feature extraction) and fusion rule design (classification). The original image content are the input of the CNN model. Thus, the network in this study should be deeper than the "shallow" network used in [19]. Considering that the GPU parallel computation is becoming more and more popular, the computational speed of CNN-based fusion is not a concern nowadays. In addition, owing to the convolutional characteristic of CNNs [39], the source images in our method can be fed to the network as a whole to further improve the computational efficiency.

The rest of this paper is organized as follows. In Section 2, we give a brief introduction to CNN and explain its feasibility as well as advantage for image fusion problem. In Section 3, the proposed CNN-based multi-focus fusion method is presented in detail. The experimental results and discussions are provided in Section 4. Finally, Section 5 concludes the paper.

2. CNN model for image fusion

2.1. CNN model

CNN is a typical deep learning model, which attempts to learn a hierarchical feature representation mechanism for signal/image data with different levels of abstraction [40]. More concretely, CNN is a trainable multi-stage feed-forward artificial neural network and each stage contains a certain number of *feature maps* corresponding to a level of abstraction for features. Each unit or coefficient in a feature map is called a *neuron*. The operations such as linear convolution, non-linear activation and spatial pooling applied to neurons are used to connect the feature maps at different stages.

Local receptive fields, shared weights and *sub-sampling* are three basic architectural ideas of CNNs [38]. The first one indicates a neuron at a certain stage is only connected with a few spatially neighboring neurons at its previous stage, which is in accord with the mechanism of mammal visual cortex. As a result, local convolutional operation is performed on the input neurons in CNNs, unlike the fully-connected mechanism used in conventional multilayer perception. The second idea means the weights of a convolutional kernel is spatially invariant in feature maps at a certain stage. By combining these two ideas, the number of weights to be trained is greatly reduced. Mathematically, let x^i and y^j denote the i-th input feature map and j-th output feature map of a convolu-

tional layer, respectively. The 3D convolution and non-linear ReLU activation [41] applied in CNNs are jointly expressed as

$$y^j = \max(0, b^j + \sum_i k^{ij} * x^i), \quad (1)$$

where k^{ij} is the convolutional kernel between x^i and y^j, and b^j is the bias. The symbol $*$ indicates convolutional operation. When there are M input maps and N output maps, this layer will contain N 3D kernels of size $d \times d \times M$ ($d \times d$ is the size of local receptive fields) and each kernel owns a bias. The last idea sub-sampling is also known as pooling, which can reduce data dimension. Max-pooling and average-pooling are popular operations in CNNs. As an example, the max-pooling operation is formulated as

$$y^i_{r,c} = \max_{0 \le m,n < s} \{x^i_{r \cdot s+m, c \cdot s+n}\}, \quad (2)$$

where $y^i_{r,c}$ is the neuron located at (r, c) in the i-th output map of a max-pooing layer, and it is assigned with the maximal value over a local region of size $s \times s$ in the i-th input map x^i. By combining the above three ideas, convolutional networks could obtain some important invariances on translation and scale to a certain degree.

In [42], Krizhevsky et al. proposed a CNN model for image classification and achieved a landmark success. In the past three years, CNNs have been successfully introduced into various fields in computer vision from high-level tasks to low-level tasks, such as face detection [43], face recognition [44], semantic segmentation [45], super-resolution [46], patch similarity comparison [47], etc. These CNN-based methods usually outperform conventional methods in their respective fields, owning to the fast development of modern powerful GPUs, the great progress on effective training techniques, and the easy access to a large amount of image data. This study also benefits from these factors.

2.2. CNNs for image fusion

2.2.1. Feasibility

As mentioned above, the generation of focus map in image fusion can be viewed as a classification problem [19]. Specifically, the activity level measurement is known as feature extraction, while the role of fusion rule is similar to that of a classifier used in general classification tasks. Thus, it is theoretically feasible to employ CNNs for image fusion. The CNN architecture for visual classification is an end-to-end framework [38], in which The input is an image while the output is a label vector that indicates the probability for each category. Between these two ends, the network consists of several convolutional layers (a non-linear layer like ReLU always follows a convolutional layer, so we don't explicitly mention it later), max-pooling layers and fully-connected layers. The convolutional and max-pooling layers are generally viewed as feature extraction part in the system, while the fully-connected layers existing at the output end are regarded as the classification part.

We further explain this point from the view of implementation. For most existing fusion methods, either in spatial domain or transform domain, the activity level measurement is essentially implemented by designing local filters to extract high-frequency details. On one hand, for most transform domain fusion methods, the images or image patches are represented using a set of predesigned bases such as wavelet or trained dictionary atoms. From the view of image processing, this is generally equivalent to convolving them with those bases [46]. For example, the implementation of discrete wavelet transform is exactly based on filtering. On the other hand, for spatial domain fusion methods, the situation is even clearer that so many activity level measurements are based on high-pass spatial filtering. Furthermore, the fusion rule, which is usually interpreted as the weight assignment strategy for different source images based on the calculated activity level measures, can be transformed into a filtering-based form as well. Considering that the basic operation in a CNN model is convolution (the full connection operation can be viewed as convolution with the kernel size that equals to the spatial size of input data [45]), it is practically feasible to apply CNNs to image fusion.

2.2.2. Superiority

Similar to the situation in visual object classification applications, the advantages of CNN-based fusion method over existing methods are twofold. First, it overcomes the difficulty on manually designing complicated activity level measurement and fusion rules. The main task is replaced by the design of network architecture. With the emergence of some easy-to-use CNN platforms such as Caffe [48] and MatConvNet [49], the implementation of network design becomes convenient to researchers. Second, and more importantly, the activity level measurement and fusion rule can be jointly generated via learning a CNN model. The learned result can be viewed as an "optimal" solution to some extent, and therefore is likely to be more effective than manually designed ones. Thus, the CNN-based method has a great potential to produce fusion results in higher quality than conventional methods.

3. The proposed method

3.1. Overview

In this Section, the proposed CNN-based multi-focus image fusion method is presented in detail. The schematic diagram of our algorithm is shown in Fig. 1. In this study, we mainly consider the situation that there are only two pre-registered source images. To deal with more than two multi-focus images, one can fuse them one by one in series. It can be seen from Fig. 1 that our method consists of four steps: *focus detection, initial segmentation, consistency verification* and *fusion*. In the first step, the two source images are fed to a pre-trained CNN model to output a score map, which contains the focus information of source images. Particularly, each coefficient in the score map indicates the focus property of a pair of corresponding patches from two source images. Then, a focus map with the same size of source images is obtained from the score map by averaging the overlapping patches. In the second step, the focus map is segmented into a binary map with a threshold of 0.5. In the third step, we refine the binary segmented map with two popular consistency verification strategies, namely, small region removal and guided image filtering [50], to generate the final decision map. In the last step, the fused image is obtained with the final decision map using the pixel-wise weighted-average strategy.

3.2. Network design

In this work, multi-focus image fusion is viewed as a two-class classification problem. For a pair of image patches $\{p_A, p_B\}$ of the same scene, our goal is to learn a CNN whose output is a scalar ranging from 0 to 1. Specifically, the output value should be close to 1 when p_A is focused while p_B is defocused, and the value should be close to 0 when p_B is defocused while p_A is focused. In other words, the output value indicates the focus property of the patch pair. To this end, we employ a large number of patch pairs as training examples. Each training example is a patch pair of the same scene. One training example $\{p_1, p_2\}$ is defined as a positive example when p_1 is clearer than p_2, and its label is set to 1. On the contrary, the example is defined as a negative example when p_2 is clearer than p_1 and the label is set to 0.

In practical usage, the source images have arbitrary spatial size. One possible way is to apply sliding-window technique to divide the images into overlapping patches, and then input each pair of

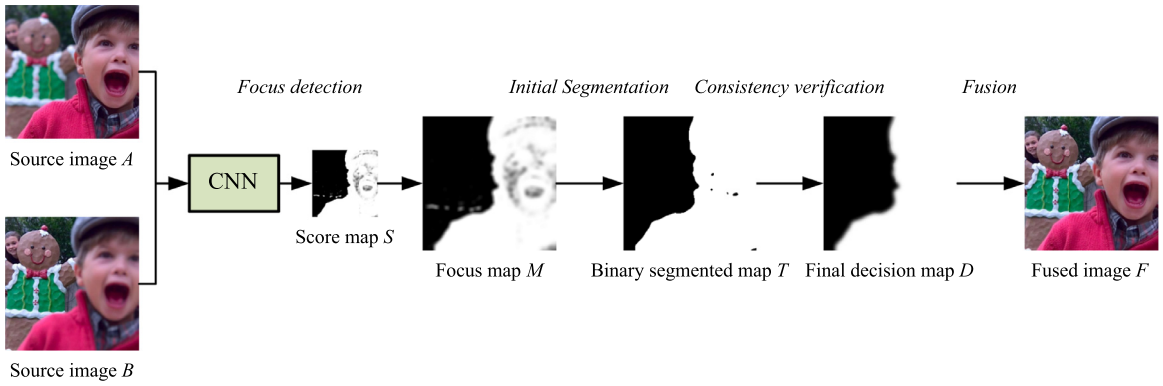


Fig. 1. Schematic diagram of the proposed CNN-based multi-focus image fusion algorithm. Data courtesy of M. Nejati [30].

patches into the network to obtain a score. However, considering that there are a large number of repeated convolutional calculations since the patches are greatly overlapped, this patch-based manner is very time consuming. Another approach is to input the source images into the network as a whole without dividing them into patches, as was applied in [39,43,45], aiming to directly generate a dense prediction map. Since the fully-connected layers have fixed dimensions on input and output data, to make it possible, the fully-connected layers should be firstly converted into convolutional layers by reshaping parameters [39,43,45] (as mentioned above, the full connection operation can be viewed as convolution with the kernel size that equals to the spatial size of input data [45], so the offline reshaping process is straightforward). After the conversion, the network only consists of convolutional and max-pooling layers, so it can process source images of arbitrary size as a whole to generate dense predictions [39]. As a result, the output of the network now is a score map, and each coefficient within it indicates the focus property of a pair of patches in source images. The patch size equals to the size of training examples. When the kernel stride of each convolutional layer is one pixel, the stride of adjacent patches in source images will be just determined by the number of max-pooling layers in the network. To be more specific, the stride is 2^k when there are totally k max-pooling layers and each with a kernel stride of two pixels [39,43,45].

In [47], three types of CNN models are presented for patch similarity comparison: *siamese, pseudo-siamese* and *2-channel*. The siamese network and pseudo-siamese network both have two branches with the same architectures, and each branch takes one image patch as input. The difference between these two networks is the two branches in the former one share the same weights while in the latter one do not. Thus, the pseudo-siamese network is more flexible than the siamese one. In the 2-channel network, the two patches are concatenated as a 2-channel image to be fed to the network. The 2-channel network just has one trunk without branches. Clearly, for any solution of a siamese or pseudo-siamese network, it can be reshaped to the 2-channel manner, so the 2-channel network provides further more flexibility [47]. All the above three types of networks can be adopted in the proposed CNN-based image fusion method. In this work, we choose the siamese one as our CNN model mainly for the following two considerations. First, the siamese network is more natural to be explained in image fusion tasks. The two branches with same weights demonstrate that the approach of feature extraction or activity level measure is exactly the same for two source images, which is a generally recognized manner in most image fusion methods. Second, a siamese network is usually easier to be trained than the other two types of networks. As mentioned above, the siamese network can be viewed as a special case of the pseudo-siamese one and 2-channel one, so its solution space is much smaller than those of the other two types, leading to an easier convergence.

Another important issue in network design is the selection of input patch size. When the patch size is set to 32×32, the classification accuracy of the network is usually higher since more image contents are used. However, there are several defects which cannot be ignored using this setting. As is well known, the max-pooling layers have important significance to the performance of a convolutional network. When the patch size is 32×32, the number of max-pooling layers is not easy to determine. More specifically, when there are two or even more max-pooling layers in a branch, which means that the stride of patches is at least four pixels, the fusion results tend to suffer from block artifacts. On the other hand, when there is only one max-pooling layer in a branch, the CNN model size is usually very large since the number of weights in fully-connected layers significantly increases. Furthermore, for multi-focus image fusion, the setting of 32×32 is often not very accurate because a 32×32 patch is more likely to contain both focused and defocused regions, which will lead to undesirable results around the boundary regions in the fused image. When the patch size is set to 8×8, the patches used to train a CNN model is too small that the classification accuracy cannot be guaranteed. Based on the above considerations as well as experimental tests, we set the patch size to 16×16 in this study.

Fig. 2 shows the CNN model used in the proposed fusion algorithm. It can be seen that each branch in the network has three convolutional layers and one max-pooling layer. The kernel size and stride of each convolutional layer are set to 3×3 and 1, respectively. The kernel size and stride of the max-pooling layer are set to 2×2 and 2, respectively. The 256 feature maps obtained by each branch are concatenated and then fully-connected with a 256-dimensional feature vector. The output of the network is a 2-dimensional vector that is fully-connected with the 256-dimensional vector. Actually, the 2-dimensional vector is fed to a 2-way softmax layer (not shown in Fig. 2) which produces a probability distribution over two classes. In the test/fusion process, after converting the two fully-connected layers into convolutional ones, the network can be fed with two source images of arbitrary size as a whole to generate a dense score map [39,43,45]. When the source images are of size $H \times W$, the size of the output score map is $(\lceil H/2 \rceil - 8 + 1) \times (\lceil W/2 \rceil - 8 + 1)$, where $\lceil \cdot \rceil$ denotes the ceiling operation. Fig. 3 shows the correspondence between the source images and the obtained score map. Each coefficient in the score map keeps the output score of a pair of source image patches of size 16×16 going forward through the network. In addition, the stride of the adjacent patches in source images is two pixels because there is one max-pooling layer in each branch of the network.

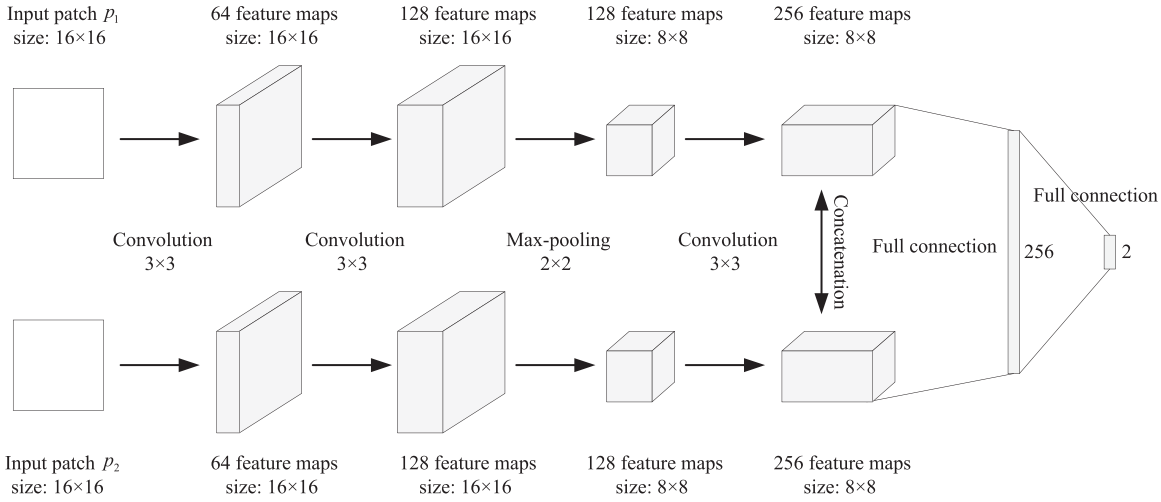


Fig. 2. The CNN model used in the proposed fusion algorithm. Please notice that the spatial size marked in the figure just indicates the training process. In the test/fusion process, after converting the two fully-connected layers into convolutional ones, the network could process source images of arbitrary size as a whole without dividing them into patches.

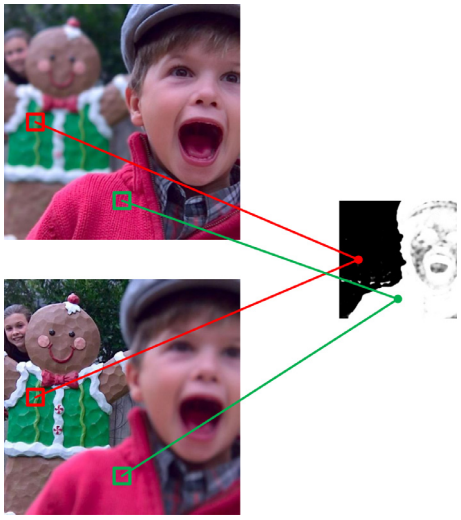


Fig. 3. The correspondence between the source images and the obtained score map.

3.3. Training

The training examples are generated from the images in ILSVRC 2012 validation image set, which contains 50,000 high-quality natural images deriving from the ImageNet dataset [51]. For each image (converted into grayscale space at first), five blurred versions with different blurring level are obtained using Gaussian filtering. Specifically, a Gaussian filter with a standard deviation of 2 and cut off to 7×7 is adopted here. The first blurred image is obtained from the original clear image with the Gaussian filter. The second blurred image is obtained from the first blurred image with the filter, and so on. Then, for each blurred image and the original image, 20 pairs of patches of size 16×16 are randomly sampled (the patch sampled from the original image must has a variance larger than a threshold, e.g., 25). In this study, we totally obtain 1,000,000 pairs of patches from the dataset (only about 10,000 images are used). Let p_c and p_b denote a pair of clear and blurred patches, respectively. It is defined as a positive example (label is set to 1) when $p_1 = p_c$ and $p_2 = p_b$, where p_1 and p_2 are the input of the first and second branch (in accord with the definition in Section 3.2 and Fig. 2), respectively. On the contrary, it is defined as a negative example (label is set to 0) when $p_1 = p_b$ and $p_2 = p_c$. Thus, the training set finally consists of 1,000,000 positive examples and 1,000,000 negative examples.

As with CNN-based classification tasks [42–44], the softmax loss function (multinomial logistic loss of the output after applying softmax) is used as the objective of our network. The stochastic gradient descent (SGD) is applied to minimize the loss function. In our training procedure, the batch size is set to 128. The momentum and the weight decay are set to 0.9 and 0.0005, respectively. The weights are updated with the following rule

$$v_{i+1} = 0.9 \cdot v_i - 0.0005 \cdot \alpha \cdot w_i - \alpha \cdot \frac{\partial L}{\partial w_i}, \qquad w_{i+1} = w_i + v_{i+1}, \tag{3}$$

where v is the momentum variable, i is the iteration index, α is the learning rate, L is the loss function, and $\frac{\partial L}{\partial w_i}$ is the derivative of the loss with respect to the weights at w_i. We train our CNN model using the popular deep learning framework Caffe [48]. The weights of each convolutional layer are initialized with the Xavier algorithm [52], which adaptively determines the scale of initialization according to the number of input and output neurons. The biases in each layer are initialized as 0. The leaning rate is equal for all layers and initially set to 0.0001. We manually drop it by a factor of 10 when the loss reaches a stable state. The trained network is finally obtained after about 10 epochs through the 2 million training examples. The learning rate is dropped one time throughout the training process.

One may notice that the training examples could be sampled from real multi-focus image dataset rather than just artificially created via Gaussian filtering. Of course, this idea is good and feasible. Actually, we experimentally verify this idea by building another training set in which half of the examples originate from a real multi-focus image set while the other half are still obtained by the Gaussian filtering based approach. We also construct a validation set which contains 10,000 patch pairs from some other multi-focus images for verification. The result shows that the classification accuracies using the above two training set with same training process are approximately the same, both around 99.5% (99.49% for the pure Gaussian filtering based set while 99.52% for the mixed set). Moreover, from the viewpoint of final image fusion results, the difference between these two approaches is even smaller that can be neglected. This test indicates that the classifier trained by the Gaussian filtering based examples can tackle the defocus blur

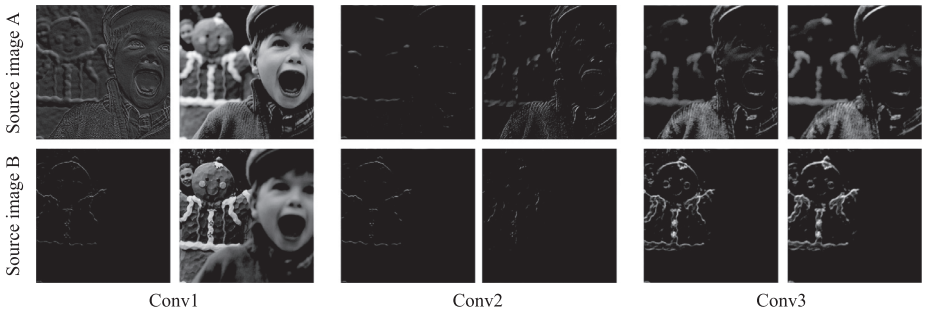


Fig. 4. Some representative output feature maps of the each convolutional layer. "conv1", "conv2" and "conv3" denote the first, second and third convolutional layer, respectively.

very well. An explanation about it is that in our opinion, as the Gaussian blur is conducted on five different standard deviations, the trained classifier could handle most blur situations, which is not limited to the situations of five discrete standard deviations in the training set, but greatly expended to a lot of combinations (may be linear or nonlinear) of them. Therefore, there is a very large possibility to cover the situations of defocus blur in multi-focus photography. To verify it, we apply a new training set which consists of Gaussian filtered examples using only three different standard deviations, and the corresponding classification accuracy on the validation set has a remarkable decrease to 96.7%. Further study on this point could be performed in the future. In this work, we just employ the above pure Gaussian filtering based training set. Furthermore, there is one benefit when using this artificially created training set. That is, we can naturally extend the learned CNN model to other-type image fusion issues, such as multi-modal image fusion and multi-exposure image fusion. Otherwise, when the training set contains examples sampled from multi-focus images, this extension seems to be not reasonable. Thus, the model learned from artificially created examples tends to have a stronger ability of generalization. In Section 4.3, we will exhibit the potential of the learned CNN model for other-type image fusion issues.

To have some insights into the learned CNN model, we provide some representative output feature maps of the each convolutional layer. The example images shown in Fig. 1 are used as the inputs. For each convolutional layer, two pairs of corresponding feature maps (the indices of two branches are the same) are shown in Fig. 4. The values of each map are normalized to the range of [0, 1]. For the first convolutional layer, some feature maps captures high-frequency information as shown in the left column while some others are similar to the input images as shown in the right column. This indicates the spatial details cannot be fully characterized by the first layer. The feature maps of the second convolutional layers mainly concentrate on the extraction of spatial details covering various gradient orientations. As shown in Fig. 4, the left and right columns mainly capture horizontal and vertical gradient information, respectively. These gradient information are integrated by the third convolutional layer, as its output feature maps successfully characterize the focus information of different source images. Accordingly, with the following two fully-connected layers, an accurate score map could be finally obtained.

3.4. Detailed fusion scheme

3.4.1. Focus detection

Let A and B denote the two source images. In the proposed fusion algorithm, the source images are converted to grayscale space if they are color images. Let $\hat{A}$ and $\hat{B}$ denote the grayscale version of A and B (keep $\hat{A} = A$ and $\hat{B} = B$ when the source images are originally in grayscale space), respectively. A score map S is obtained by feeding $\hat{A}$ and $\hat{B}$ to the trained CNN model. The value of each coefficient in S ranges from 0 to 1, which suggests the focus property of a pair of patches of size 16×16 in source images (see Fig. 3). The closer the value is to 1 or 0, the more focused the patch from source image $\hat{A}$ or $\hat{B}$ is. For two neighboring coefficients in S, their corresponding patches in each source image are overlapped with a stride of two pixels. To generate a focus map (denoted as M) with the same size of source images, we assign the value of each coefficient in S to all the pixels within its corresponding patch in M and average the overlapping pixels. Fig. 5(a) shows the obtained focus map of the example illustrated in Fig. 1. It can be seen that the focus information is accurately detected. Intuitively, the values of the regions with abundant details seems to be close to 1 (white) or 0 (black), while the plain regions tend to own values close to 0.5 (gray).

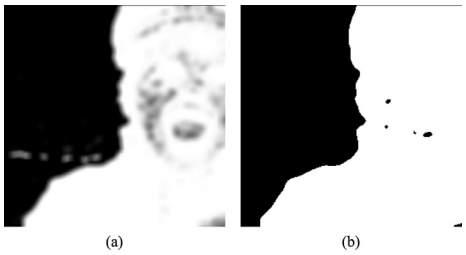


Fig. 5. Initial segmentation. (a) Focus map (b) binary segmentation map.

3.4.2. Initial segmentation

To preserve useful information as much as possible, the focus map M needs to be further processed . In our method, as with most spatial domain multi-focus image fusion methods [18–22,24–27,29,30], we also adopt the popular "choose-max" strategy to process M. Accordingly, a fixed threshold of 0.5 is applied to segment M is into a binary map T, which is in accord with the classification principle of the learned CNN model. That is, the focus map is segmented by

$$T(x,y) = \begin{cases} 1, & M(x,y) > 0.5 \\ 0, & otherwise \end{cases}. \tag{4}$$

The obtained binary map is shown in Fig. 5(b) (please notice the optical illusion in the focus map shown in Fig. 5(a), namely, the gray regions seems to be darker than its real intensity in a white background while brighter than its real intensity in a black background). It can be seen that almost all the gray pixels in the focus map are correctly classified, which demonstrates that the learned CNN model can obtain precise performance even for the plain regions in source images.

3.4.3. Consistency verification

It can be seen from Fig. 5(b) that the binary segmented map is likely to contain some mis-classified pixels, which can be eas-

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

Fig. 6. Consistency verification and fusion. (a) Initial decision map (b) Initial fused image (c) final decision map (d) fused image.

ily removed using the small region removal strategy. Specifically, a region which is smaller than an area threshold is reversed in the binary map. One may notice that the source images sometimes happen to contain very small holes. When this rare situation occurs, users can manually adjust the threshold even to zero, which means the region removal strategy is not applied. We will show in the next Section that the binary classification results can already achieve high accuracy. In this paper, the area threshold is universally set to $0.01 \times H \times W$, where H and W are the height and width of each source image, respectively. Fig. 6(a) shows the obtained initial decision map after applying this strategy.

Fig. 6(b) shows the fused image using the initial decision map with the weighted-average rule. It can be seen that there are some undesirable artifacts around the boundaries between focused and defocused regions. Similar to [30], we also take advantage of the guided filter to improve the quality of initial decision map. Guided filter is a very efficient edge-preserving filter, which can transfer the structural information of a guidance image into the filtering result of the input image. The initial fused image is employed as the guidance image to guide the filtering of initial decision map. There are two free parameters in the guided filtering algorithm: the local window radius r and the regularization parameter ε. In this work, we experimentally set r to 8 and ε to 0.1. Fig. 6(c) shows the filtering result of the initial decision map given in Fig. 6(b).

3.4.4. Fusion

Finally, with the obtained decision map D, we calculate the fused image F with the following pixel-wise weighted-average rule

$$F(x,y) = D(x,y)A(x,y) + (1 - D(x,y))B(x,y). \quad (5)$$

The fused image of the given example is shown in Fig. 6(d).

4. Experiments

4.1. Experimental settings

To verify the effectiveness of the proposed CNN-based fusion method, 40 pairs of multi-focus images are used in our experiments. 20 pairs among them have been widely employed in multi-focus image fusion research, while the other 20 pairs come from a new multi-focus image dataset "Lytro" which is publicly available online [53]. A portion of the test image set is shown in Fig. 7, where the first two rows list eight traditional image pairs and the last two rows list ten image pairs of the new dataset.

The proposed fusion method is compared with six representative multi-focus image fusion methods, which are the non-subsampled contourlet transform (NSCT)-based one [6], the sparse representation (SR)-based one [13], the NSCT-SR-based one [11], the guided filtering (GF)-based one [25], the multi-scale weighted gradient (MWG)-based one [27] and the dense SIFT (DSIFT)-based one [29]. Among them, the NSCT-based, SR-based and NSCT-SR-based methods belong to transform domain methods. The NSCT-based method is verified to be able to outperform most MST-based methods in multi-focus image fusion [54]. The SR-based method using the sliding window technique owns advantages over conventional MST-based methods. The recently introduced NSCT-SR-based fusion method is capable of overcoming the respective defects of the NSCT-based method and the SR-based method. The GF-based, MWG-based and DSIFT-based methods are all recently proposed spatial domain methods with elaborately designed fusion schemes, which are commonly recognized to generate state-of-the-art results in the field of multi-focus image fusion. In our experiments, the NSCT-based, SR-based and NSCT-SR-based methods are implemented with our MST-SR fusion toolbox available online [55], in which the related parameters are set to the recommended values introduced in related publications. The free parameters of the GF-based, MWG-based and DSIFT-based methods are all set to the default values reported in related publications. The MATLAB implementations of these three fusion methods are all available online [55–57].

Objective evaluation plays an important role in image fusion as the performance of a fusion method is mainly assessed by the quantitative scores on multiple metrics [58]. A variety of fusion metrics have been proposed in recent years. A good survey provided by Liu et al. [59] classifies them into four groups: information theory-based ones, image feature-based ones, image structural similarity-based ones and human perception-based ones. In this study, we select one metric from each category to make a comprehensive evaluation. The four selected metrics are: 1) Normalized mutual information Q_{MI} [60], which measures the amount of mutual information between fused image and source images. 2) Gradient-based metric Q_G (known as $Q^{AB/F}$ as well) [61], which assesses the extent of spatial details injected into the fused image from source images. 3) Structural similarity-based metric Q_Y proposed by Yang et al. [62], which measures the amount of structural information preserved in the fused image. 4) Human perception-based metric Q_{CB} proposed by Chen and Blum [63], which addresses the major features in human visual system. For each of the above four metrics, a larger value indicates a better fusion performance.

4.2. Experimental results and discussions

4.2.1. Commutativity verification

One fundamental rule in image fusion is commutativity, which means that the order of source images make no difference to the fusion result. Considering the designed network, although the two branches share the same weights, the commutativity of the network seems to be not valid since there exist fully-connected layers. When the source images are switched, the output of fully-connected layers may not be switched accordingly to ensure commutativity. Fortunately, a training technique used in Section 3.3 makes the fusion algorithm approximately commutable at a very high level, leading to no influence of switching order on the fusion result. We first experimentally verify this point and then explain the reason behind it.

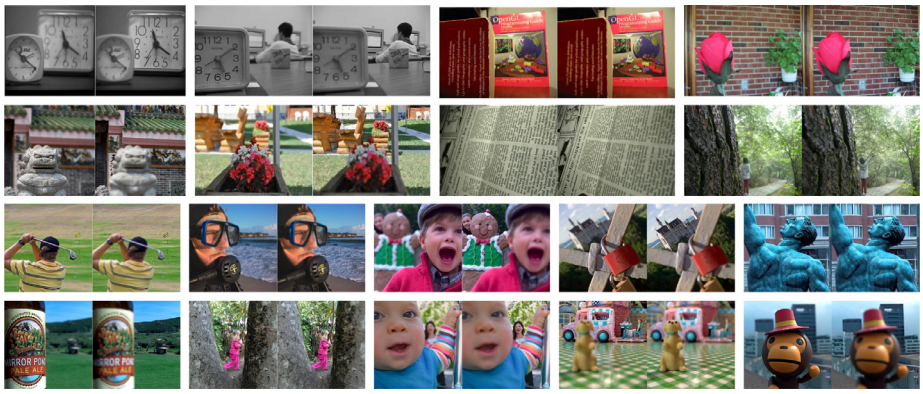


Fig. 7. A portion of multi-focus test images used in our experiments.

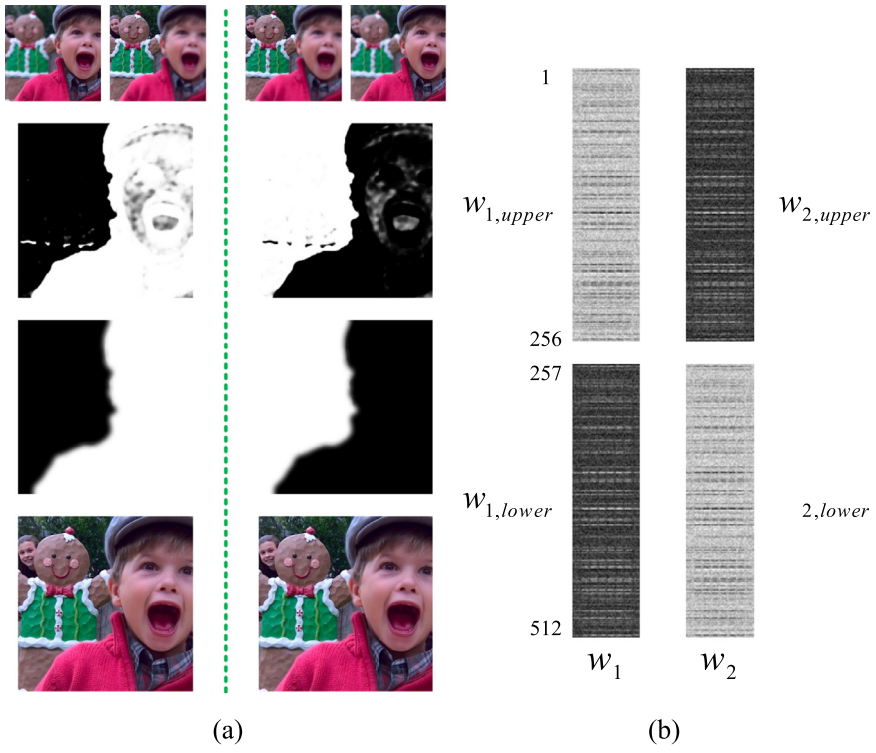


Fig. 8. Commutativity verification of the proposed algorithm. (a) A fusion example with two different input orders for source images. (b) the illustration of weight vectors of a learned CNN model.

For all the 40 pairs of source images, we fuse them using the proposed algorithm with two possible input orders. Thus, two sets of results are obtained. We use both the score map and the final fused image to verify the commutativity. For each score map pair from the same source images, a pixel-wise summation is performed. Ideally, the sum value at each pixel is 1 if the commutativity is valid. To verify it, we calculate the average value over all the pixels in the sum map. The mean and standard deviation of the above average values over all the 40 examples are 1.001468 and 0.001873, respectively. We also use the SSIM index [64] to measure the closeness between two fused images of the same source image pairs. The SSIM score of two images is 1 when they are identical. The mean and standard deviation of SSIM scores over all the 40 examples are 0.999964 and 0.000161, respectively. The above results demonstrate that the proposed method almost owns ideal commutativity. Fig. 8(a) shows an example of this verification. The two columns shown in Fig. 8(a) demonstrate the fusion results with two different input orders for source images. The score maps, decision maps and fused images are shown in the second, third and fourth rows, respectively. It is clear that the commutativity is valid.

For simplicity, we apply a slight CNN model to explain the above results. The first fully-connected layer is removed from the network, so the 512 feature maps after concatenation are directly connected to a 2-dimensional output vector (please refer to Section 4.2.4 for more details of this slight model). The slight model is trained with the same approach as the original model. Let w_1 and w_2 denote the weight vector connected to the first (upper) and second (lower) neurons, respectively. The vector dimension is $512 \times 64 = 32768$, in which 64 indicates the amount of coefficients in each feature map of size 8×8. As shown in Fig. 8(b), w_1 and w_2 are displayed in a 512×64 matrix form being divided into two parts of the same size: upper part and lower part. Each part has a size of 256×64. This dividing is meaningful since the 512 feature maps are concatenated from two branches. As shown in Fig. 8(b), we use $w_{1,\ upper}$ and $w_{1,\ lower}$ to denote the upper and lower part of w_1, respectively. The same definition scheme is applied to w_2. Thus, $w_{1,\ upper}$ indicates the weights associated with the connection from the feature maps come from the first branch to the upper neuron. The meanings of $w_{1,\ lower}$, $w_{2,\ higher}$ and $w_{2,\ lower}$ are similar. Now, we can investigate the con-

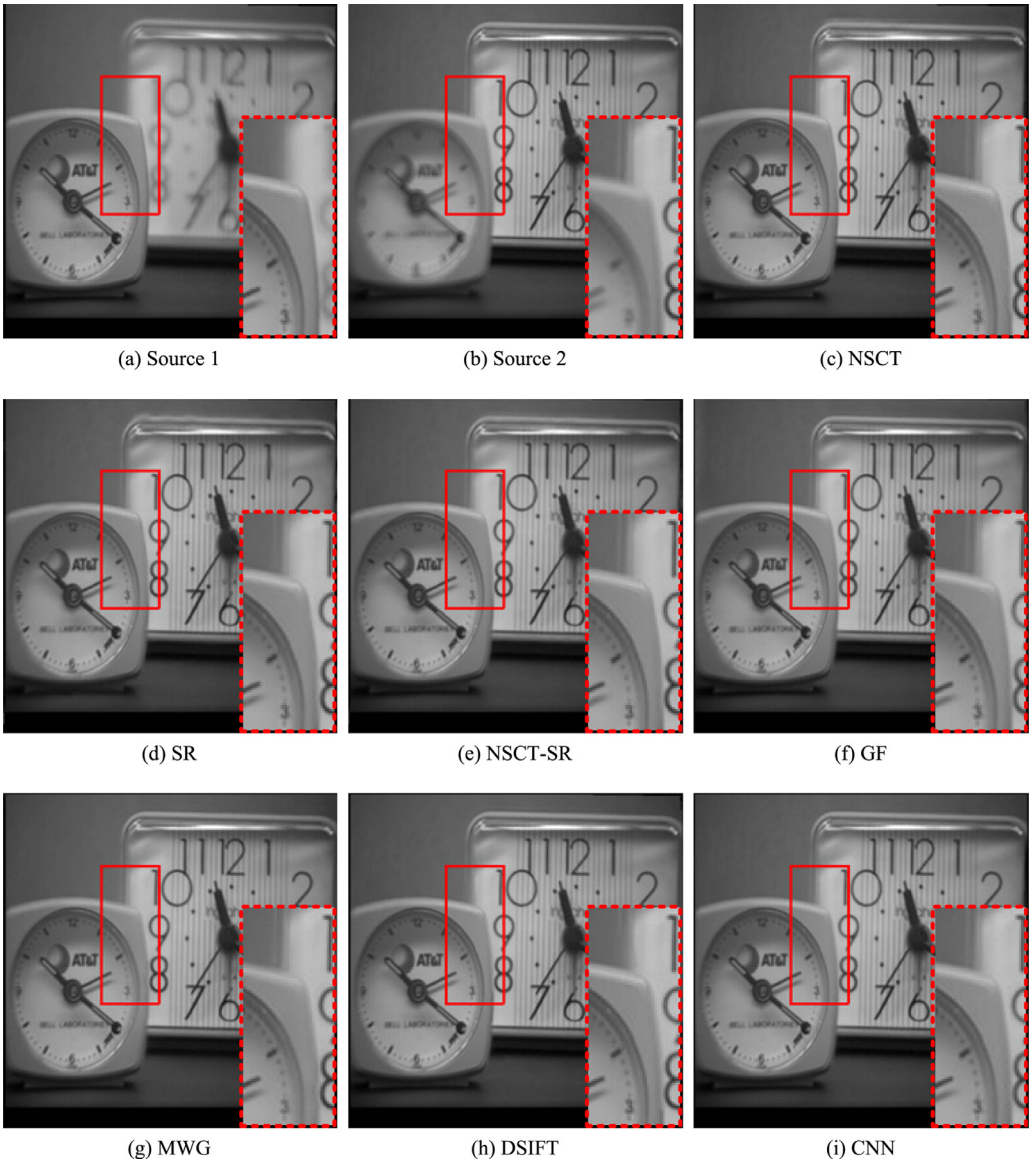


Fig. 9. The "clock" source image pair and their fused images obtained with different fusion methods.

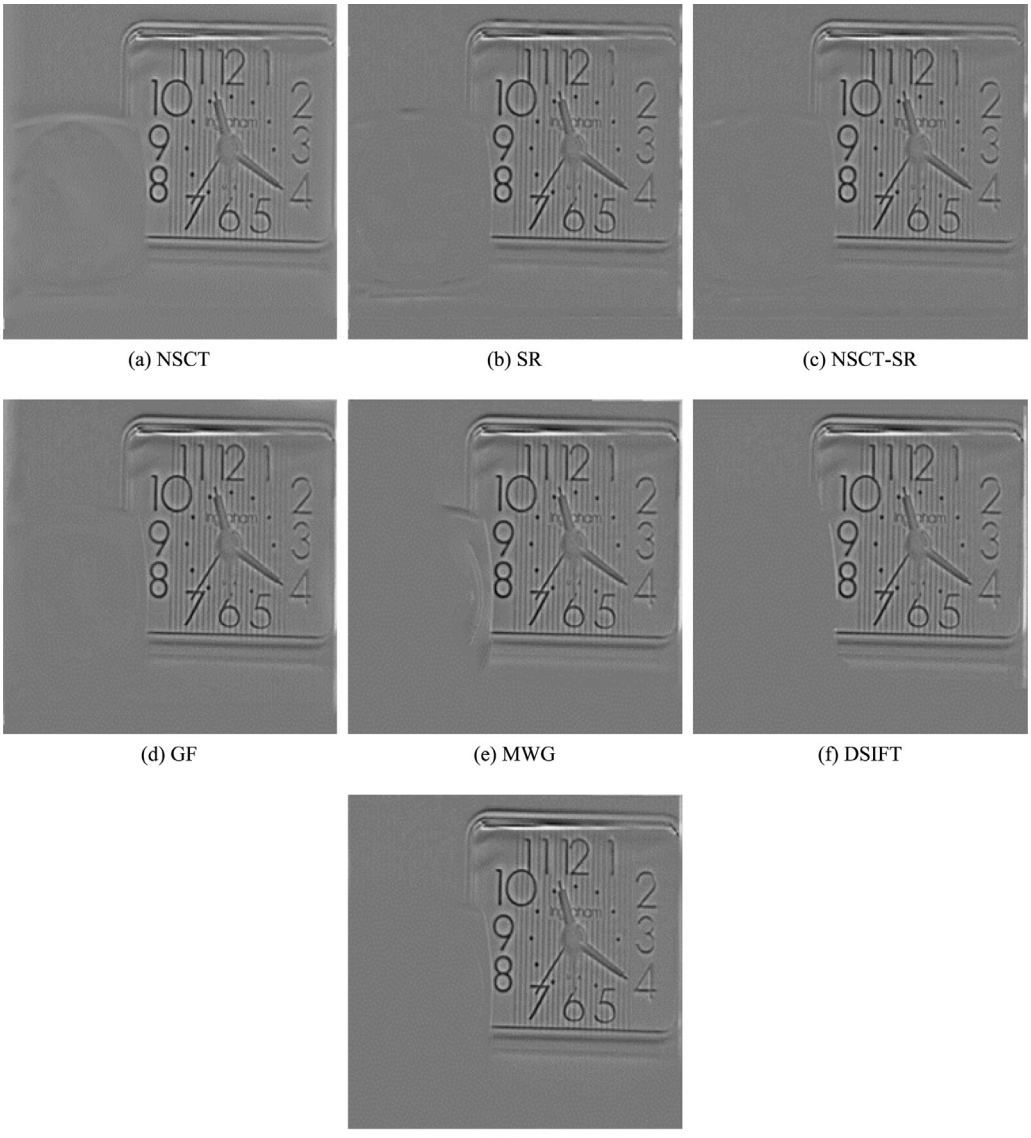
dition under which the commutativity is valid. Let v_A and v_B denote the output feature maps (As the two branches share the same weights, using either the first or the second branch leads to the same result) of two corresponding source image patches s_A and s_B, respectively. We experimentally find that the biases of both two neurons are very close to 0. Therefore, when we put s_A into the first branch while s_B into the second, the outputs of the upper and lower neurons are $V_{11} = w^T_{1,upper}v_A + w^T_{1,lower}v_B$ and $V_{12} = w^T_{2,upper}v_A + w^T_{2,lower}v_B$, respectively. On the contrary, when we put s_A into the second branch while s_B into the first, the outputs of the upper and lower neurons are $V_{21} = w^T_{1,upper}v_B + w^T_{1,lower}v_A$ and $V_{22} = w^T_{2,upper}v_B + w^T_{2,lower}v_A$, respectively. Clearly, if $w_{1,upper} = w_{2,lower}$ and $w_{1,lower} = w_{2,upper}$, there will be $V_{11} = V_{22}$ and $V_{12} = V_{21}$, namely, the commutativity is valid. In Fig. 8(b), $w_{1,\ upper}$, $w_{1,\ lower}$, $w_{2,\ upper}$ and $w_{2,\ lower}$ are shown after a normalization to the range of [0, 1]. Intuitively, it can been seen that the above condition is well satisfied. Quantitative results also verify it. The situation of the original network with two fully-connected layer is more complex, but the analysis approach is similar. Therefore, the results of the above commutative experiments could be quantitatively explained.

The last question is why the above condition is satisfied. From our perspective, this is mainly due to construction of training examples. Specifically, a positive example and a negative example are simultaneously obtained by switching the order of a clear patch and its blurred version, as mentioned in Section 3.3. As the ground truth output of a positive example is $[1 \quad 0]^T$, while of a negative one is $[0 \quad 1]^T$, which satisfies the commutativity, the final learned weights will suffer from such a constraint. Up to this point, we have demonstrate and explain the commutativity of the proposed fusion algorithm, so it is not necessary to concern the order of two source images in practical usage.

4.2.2. Comparison with other fusion methods

We first compare the performance of different fusion methods based on visual perception. For this purpose, two examples are mainly provided to exhibit the difference among different methods.

Fig. 9 shows the "clock" source image pair and their fused images obtained with different fusion methods. In each image, a region around the boundary between focused and defocused parts is magnified and shown in the lower right corner. To have a bet-



(a) NSCT (b) SR (c) NSCT-SR

(d) GF (e) MWG (f) DSIFT

(g) CNN

Fig. 10. The different image between each fused image in Fig. 10(c)-(i) and the source image in Fig. 10(a).

ter comparison, Fig. 10 shows the difference image obtained by subtracting the first source image shown in Fig. 9(a) from each fused image, and the values of each difference image are normalized to the range of 0 to 1. The fusion results of the three transform domain methods contain some undesirable artifacts on the top border of the big clock (not precisely registered), especially for the NSCT-based and SR-based methods. Moreover, the difference images shown in Fig. 10(a) and (b) demonstrate that the fusion quality in the small clock regions of the NSCT-based and SR-based methods are lower than other methods. The fused image of the GF-based method shown in Fig. 9(f) is overall in high quality except that the regions around the digits "8" and "9" in the big clock are slightly blurred. The MWG-based method performs well in mis-registered regions, but its fusion quality around the boundary regions is not very high. For example, the digit "3" in the small clock is obviously blurred in the fused image (see the magnified region in Fig. 9(g)). The difference image shown in Fig. 10(e) also reveals this point. The fused image of the DSIFT-based method extract details very well, but suffers from some artifacts around object edges. In the magnified region, the border of the small clock looks unnatural and a small part of the border of the large clock (just above the small clock) disappears in the fusion result. As shown in Fig. 9(i), our CNN-based method performs well in both boundary regions and mis-registered regions. In general, the fused image of our method owns highest visual quality among all these seven methods, which can be further verified by the difference image shown in Fig. 10(g).

Fig. 11 shows the "golf" source image pair and their fused images obtained with different fusion methods, and a magnified region is also extracted in this example. The normalized difference image between each fused image and the first source image shown in Fig. 11(a) is provided in Fig. 12. It can be seen that the NSCT-based method and SR-based method don't perform very well in the focused regions of the first source image (see the T-shirt regions in Fig. 12(a) and (b)). The fusion quality of the NSCT-SR-based method is much better in terms of this issue, but the region of the golf club behind the man's head is still not well merged. The MWG-based method does not work well in this example since the details in the triangle grass region as well as the lower left grass region are lost. In addition, the boundary regions between focused and defocused parts are also not well tackled. The DSIFT-based method extracts most details in the source images, but the merging effect

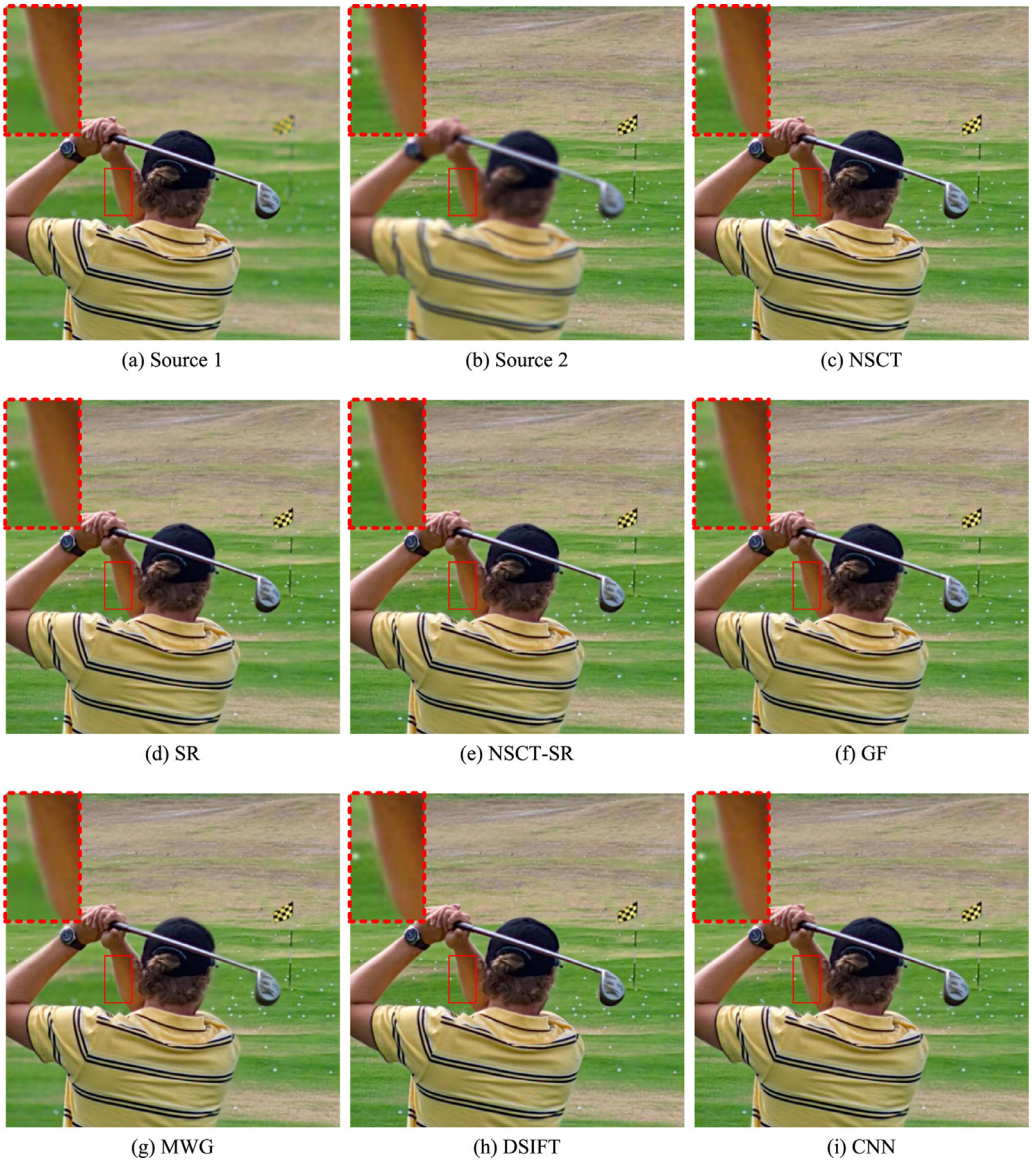


Fig. 11. The "golf" source image pair and their fused images obtained with different fusion methods.

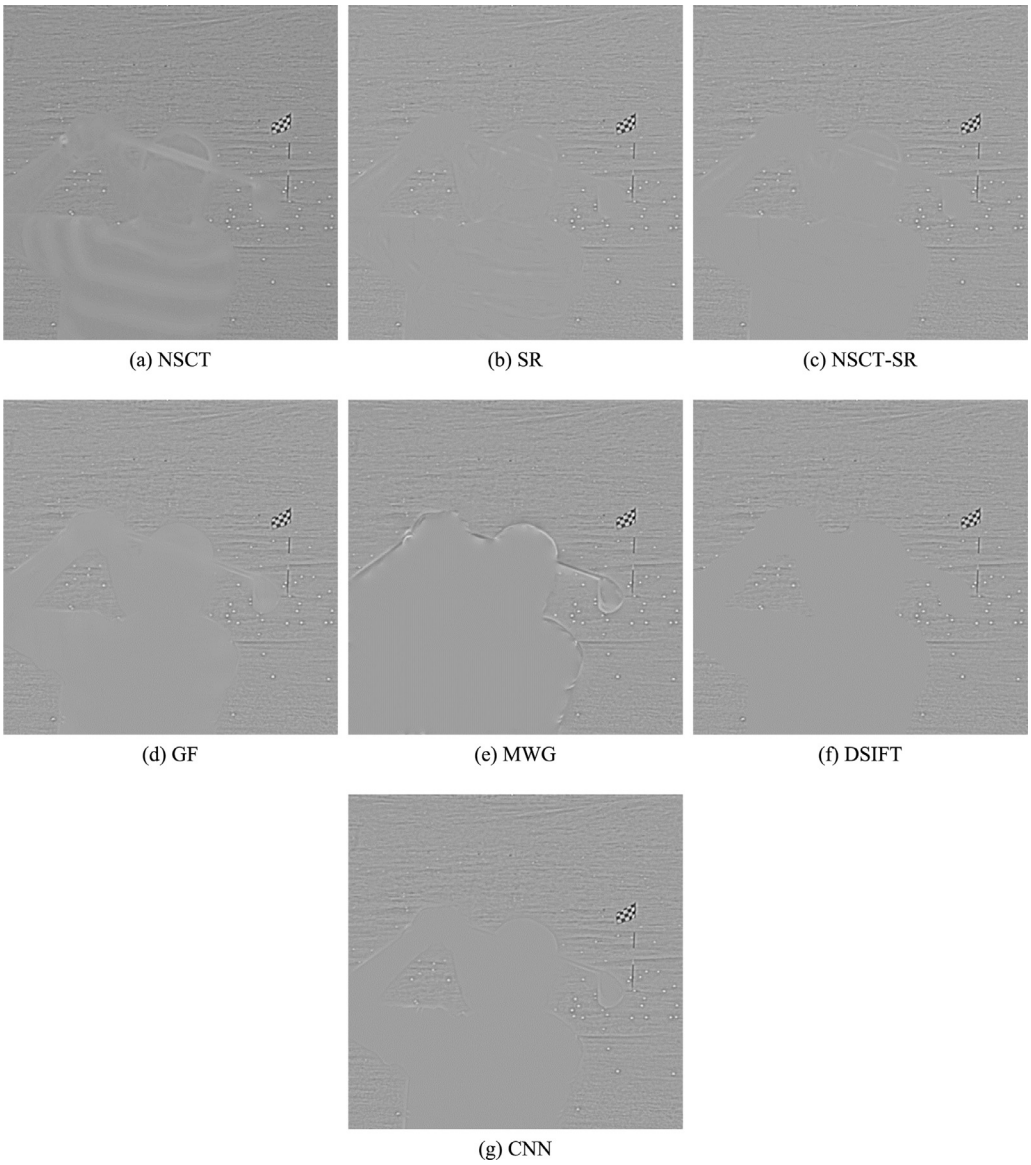
Table 1
Objective assessment of different fusion methods.

Metrics	NSCT	SR	NSCT-SR	GF	MWG	DSIFT	CNN
Q_{MI}	0.9015(0,0)	1.0798(0,2)	1.0854(0,2)	1.0632(0,0)	1.0836(1,1)	**1.1884**(37,3)	1.1455(2,32)
Q_G	0.7239(0,0)	0.7397(1,1)	0.7417(0,3)	0.7456(2,4)	0.7309(0,0)	0.7489(14,21)	**0.7497**(23,11)
Q_Y	0.9502(0,0)	0.9615(0,1)	0.9643(1,0)	0.9751(0,2)	0.9811(3,5)	0.9852(8,25)	**0.9865**(28,7)
Q_{CB}	0.7355(0,0)	0.7596(1,0)	0.7723(0,1)	0.7768(0,1)	0.7704(1,1)	0.7948(7,31)	**0.7968**(31,6)

on boundary regions is also not satisfactory (see the head of the club as well as the cap's edge in Fig. 11(h)). The boundary seems to expand outward to a little extent in the fused image of the DSIFT-based method (see Fig. 12(f)). The GF-based method and the proposed method both obtain fusion result in high quality. The difference between these two fused images is relatively small. But when carefully comparing the edge between grass and arm among all the fused images, we can see that the proposed CNN-based method produces more natural effect than all the other methods including the GF-based one.

Table 1 lists the objective performance of different fusion methods using the above four metrics. The average scores on the 40 source image pairs of each method are listed here, in which the highest value shown in bold each row denotes the best score among all the methods. The two digits within a parenthesis indicates the number of image pairs on which the corresponding method gets the first (left one) and second (right one) places, respectively.

We can see that the DSIFT-based method and the proposed method clearly beat the other five methods on the average score of each fusion metric. Moreover, the summation times that these two methods obtain the first place on Q_{MI}, Q_G, Q_Y and Q_{CB} are 39, 37, 36 and 38, respectively. The corresponding times for the second place are 35, 32, 32 and 37. The above observations indicate these two methods own obvious advantages over other methods on all the four metrics.



(a) NSCT (b) SR (c) NSCT-SR

(d) GF (e) MWG (f) DSIFT

(g) CNN

Fig. 12. The different image between each fused image in Fig. 12(c)-(i) and the source image in Fig. 12(a).

Then, we compare the DSIFT-based method with the proposed CNN-based method. It can be seen that the DSIFT-based method has a clear advantage on the metric Q_{MI}. A brief explanation about this point is provided here. According to the definition in [60], Q_{MI} measures the similarity of global statistical characteristic on gray level (via grayscale histograms) between source images and fused image, regardless of how the information is spatially distributed across the images. In [65], Hossny et al. (the authors of Q_{MI}) made a further discussion about this metric. They pointed out that Q_{MI} consistently favours simple averaging over multi-scale transform (MST)-based fusion algorithms. Furthermore, in [25], Li and Kang reported that a very high Q_{MI} value is not always a good thing because Q_{MI} tends to become larger when the pixel values of the fused image are closer to one of the source images. To verify the above two statements, we conduct a simple test on the "clock" image pair. The simple weighted average fusion with a fixed weight factor w over all the pixels can be denoted as $F(x,y) = w \cdot A(x,y) + (1-w) \cdot B(x,y)$, where A and B are two source images, and F is the fused image. We change w from 0 to 1 with a stride of 0.1. Table 2 lists the obtained Q_{MI} scores of the weighted average methods and the above seven fusion methods. It can be seen that the weighted average methods obtain the lowest score when $w = 0.5$. With the weight factor approaching to either 0 or 1, the score dramatically increases. Particularly, the score of the NSCT-based method is just a little higher than the weighted average method with $w = 0.5$, but lower than other weight settings. Moreover, when the weight factor is 1 or 0, which means that the fused image is exactly one of the source images, the obtained score is much higher than all the seven fusion methods. The situations of other source image pairs are similar. When we set the fused image as one of the source images (note that $F = A$ and $F = B$ will lead to the same Q_{MI} score), the average score over 40 source image pairs is 1.3096, which is much higher than the 1.1884

Table 2
A simple test on Q_{MI} using the "clock" image pair.

Method	$w = 0$	$w = 0.1$	$w = 0.2$	$w = 0.3$	$w = 0.4$	$w = 0.5$
Score	1.3847	1.1591	1.0794	1.0370	1.0163	0.9983
Method	$w = 0.6$	$w = 0.7$	$w = 0.8$	$w = 0.9$	$w = 1$	NSCT
Score	1.0161	1.0365	1.0804	1.1616	1.3847	1.0029
Method	SR	NSCT-SR	GF	MWG	DSIFT	CNN
Score	1.1176	1.1371	1.1031	1.1575	1.2437	1.2059

Fig. 13. Intermediate fusion results of the proposed method.

of the DSIFT-based method as listed in Table 1. Based on the above observations, it is clear that the metric Q_{MI} must be considered together with other metrics [25]. For the other three metrics Q_G, Q_Y and Q_{CB}, the proposed CNN-based method slightly outperforms the DSIFT-based method in terms of the average scores, while the advantage on the times of getting the first place is more noticeable. Overall, it can be at least stated that the proposed method obtains competitive objective performance with the DSIFT-based method.

Considering the above comparisons on subjective visual quality and objective evaluation metrics together, the proposed CNN-based fusion method can generally outperform other methods, leading to state-of-the-art performance in multi-focus image fusion.

4.2.3. Intermediate results of the proposed method

To further exhibit the effectiveness of the CNN model for multi-focus image fusion, the intermediate results of eight pairs of source images are given in Fig. 13. Looking back the proposed fusion algorithm presented in Section 3.4, the binary segmentation approach to obtain binary segmented map from the focus map with a threshold of 0.5 is in accord with the "choose-max" strategy, which is preferred in the field of multi-focus image fusion [18–22,24–27,29,30]. Therefore, for multi-focus image fusion, the binary segmented map can be interpreted as actual output of our CNN model. It can be seen from the second column of Fig. 13 that the obtained segmented maps are very accurate that most pixels are

Table 3
The average objective assessment of the proposed fusion method with and without using consistency verification (CV) techniques.

Method	Q_{MI}	Q_G	Q_Y	Q_{CB}
without CV	1.1934	0.7494	0.9859	0.7960
with CV	1.1455	0.7497	0.9865	0.7968

correctly classified, which demonstrates the good capacity of the learned CNN model.

Nevertheless, there still exist two defects in terms of the binary segmented maps. First, a few of pixels are sometimes misclassified, leading to the emergence of small regions or holes in the segmented maps. These pixels usually locate at the plain regions of the scene that the distinction between two source images are very small. Among the eight pairs of images, such situation arises in six of them except for the "golf" and "statue" set. Since those mis-classified pixels take up a very small proportion and they are usually locate at the plain regions, their impact on the fusion result is actually very slight. Even so, we apply the small region removal strategy to rectify these pixels. The third column in Fig. 13 shows obtained initial decision map after this correction. Second, the boundaries between focused and defocused regions usually suffer from slight blocking artifacts. The fourth column in Fig. 13 shows the fused images using the initial decision map. Compared with the first defect, this one is more urgent to be addressed as the fusion quality around the boundaries is more important. Fortunately, edge-preserving filtering offers an appropriate tool to solve this problem. The final decision maps shown in the fifth column of Fig. 13 obtained with the guided filter are more natural in the boundary regions, leading to high visual quality of the fused results shown in the last column of Fig. 13.

In summary, we apply two time-efficient consistency verification approaches to refine the binary classification result, namely, small region removal/filtering and guided filtering. In the literature, consistency verification has been an important role in image fusion since the 1990s [4]. Popular consistency verification approaches used in image fusion methods are usually based on various image filtering techniques, including majority filtering [4,18,19], median filtering [26], morphological filtering [22], small region filtering [22,29], edge-preserving filtering (e.g., bilateral filtering [66] and guided filtering [25]), etc. As these consistency verification techniques are usually simple and have been widely used in image fusion for many years, they can be regarded as conventional post-processing techniques. To make further progresses and pursue state-of-the-art performance, many recently proposed multi-focus image fusion methods are inclined to employ some advanced post-processing techniques, such as the image matting technique used in [26], the feature matching approach employed in [29] and the Markov Random Field (MRF)-based regularization method applied in [30]. Owing to the high focus detection accuracy of the CNN model, we don't use any complicated post-processing techniques in our fusion algorithm. Table 3 lists the objective performance of the initial fused images and the final fused images using the above four fusion metrics. The average scores on the 40 source image pairs are listed here. It can be seen that the scores has a very slight increase when applying the consistency verification techniques for Q_G, Q_Y and Q_{CB}. The situation of metric Q_{MI} is totally in accord with the analysis on this metric in Section 4.2.2. Furthermore, when considering the contents in Table 1 and Table 3 together, we can see that even the CNN-based method without using consistency verification techniques outperforms all the other fusion methods on all the four metrics, which further verifies the effectiveness of CNN for image fusion. Of course, by applying some advanced post-processing techniques as mentioned above, the fusion quality may be further improved.

Table 4
The number of parameters in each convolutional/fully-connected layer of our CNN model.

Layer	conv1	conv2	conv3	fc1	fc2	sum
Number	640	73856	295168	8388864	514	8759042

Table 5
The average objective assessment of the proposed fusion method using original and reduced models.

Model	Q_{MI}	Q_G	Q_Y	Q_{CB}
Original	1.1455	0.7497	0.9865	0.7968
Slight	1.1443	0.7482	0.9856	0.7955

Table 6
The average running time of the proposed fusion method using original and reduced CNN models for two source images of size 520 × 520 (Unit: seconds).

Model	Parallel part	Serial part	Total
Original	0.66	0.12	0.78
Slight	0.21	0.12	0.33

4.2.4. Memory consumption and computational efficiency

Table 4 lists the number of parameters (both weights and biases are involved) in each convolutional/fully-connected layer of our CNN model shown in Fig. 2. The three convolutional layers and two fully-connected layers are named "conv1", "conv2","conv3", "fc1" and "fc2" from the input to the output. As one parameter takes up 4 bytes as a single-precision floating-point variable, the size of physical memory taken by the CNN model is 35,036,168 bytes (about 33.4 MB). It is clear that the fc1 layer occupies the greatest percentage (approximately 95.8%) in the whole parameters. Therefore, a feasible approach to reduce the memory consumption greatly is removing the "fc1" layer, namely, the 512 feature maps obtained by the "conv3" layer is directly connected to a 2-dimensional output vector. In this way, the number of parameters in the remaining fully-connected layer is 65,538 (calculated as $512 \times 8 \times 8 \times 2+2$), and the total number decreases to 435,202. Thus, the slight model only takes up about 1.66 MB, which is less than one twentieth of the original model size. To test the effectiveness of the slight model for multi-focus image fusion, we replace the original model with it while the other parts in the proposed algorithm remain the same. Table 5 lists the average objective assessment over eight source image pairs of the proposed fusion method using original and slight models. It can be seen that the performance of the slight model is a little inferior to the original model, but the gap is very small. This result implies the high flexibility on CNN model design, while the network used in this work is just a feasible one.

The proposed CNN-based image fusion algorithm is implemented with C++ by calling the interfaces provided by Caffe to go forward through the network. The GPU mode is employed, so the program consists of two parts: parallel one and serial one. The parallel part denotes the procedure that source images passing through the network with Caffe interfaces to obtain the score map. The serial part denotes the subsequent procedure starting with the score map. The hardware configurations of our experiments are: a NIVIDA GeForce GTX TITAN Black GPU, an Inter Core i7-4790k CPU and 16 GB RAM. Table 6 lists the average running time of the proposed fusion method using original and slight CNN models for two source images of size 520 × 520, and the time of the parallel part and serial part are given individually. It can be seen that the

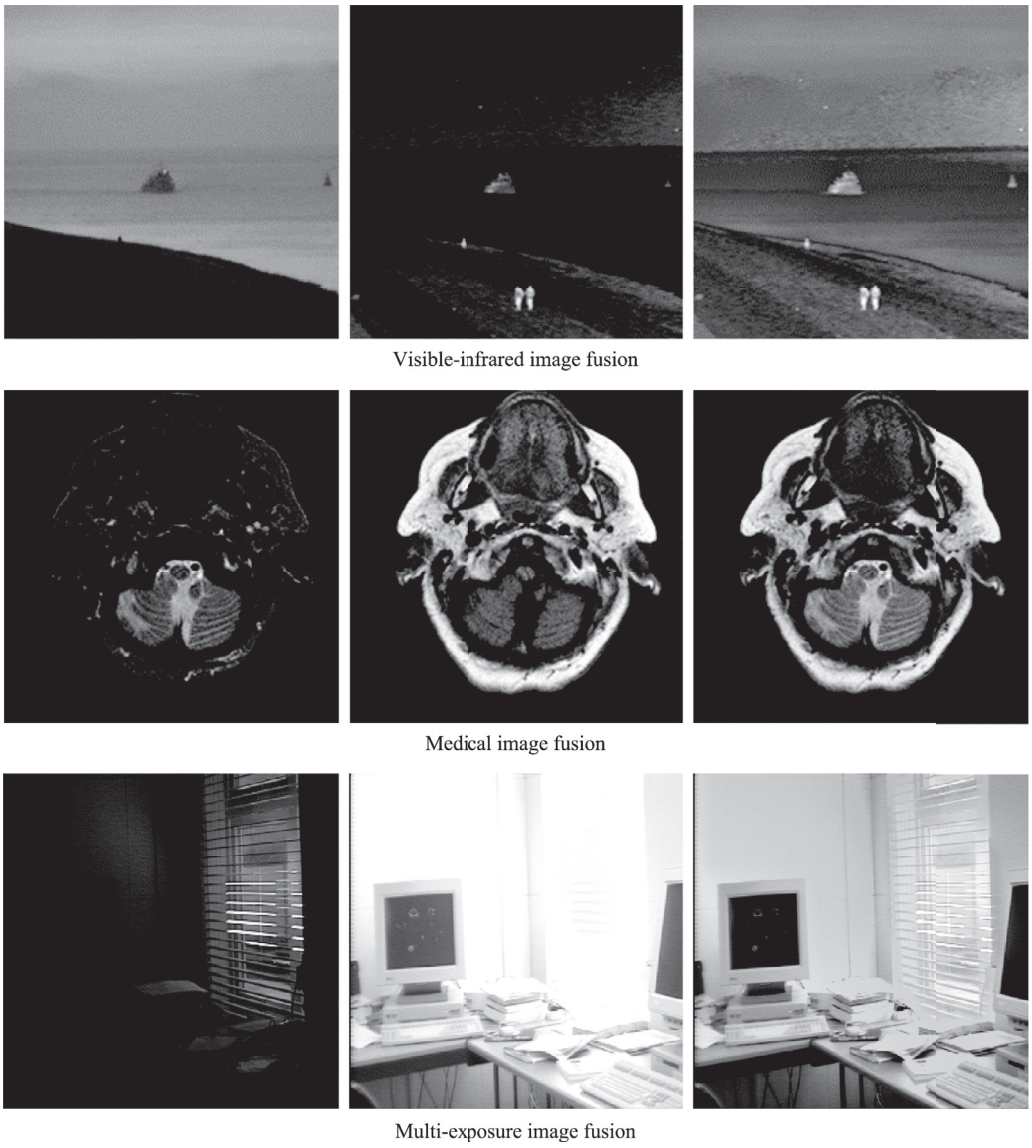


Fig. 14. Three other-type image fusion examples using the corresponding CNN-based methods presented in Section 4.3.

computing time of the proposed fusion method with the original model is less than one second, which is fast enough for practical usage. The computational efficiency using the slight model is more than two times higher than using the original model and the increasing factor of the parallel part individually is about three.

4.3. Extension to other-type image fusion issues

To exhibit the generalization ability of the learned CNN model, we extend its usage to multi-modal image fusion and multi-exposure image fusion. Since different types of image fusion issues have their own characteristics, and there exist considerable differences in terms of studying methods and basic frameworks. Like the proposed multi-focus image fusion algorithm, we just apply some commonly-used techniques to design related CNN-based methods for other-type image fusion issues.

Multi-scale transform is recognized as an effective tool in the field of multi-modal image fusion as it is consistent with human visual perception [7]. In this work, the fusion framework introduced in [67] based on Laplician pyramid (LP) and Gaussian pyramid (GP) is applied. we employ the focus map obtained from the CNN model as the weight map that indicates pixels' activity level. During the fusion process, the source images are decomposed into Laplician pyramids and the weight maps are decomposed into Gaussian pyramids. Then, at each decomposition level, the weighted average fusion rule is used to merge the information from two source images. The fused image is finally obtained via LP reconstruction.

For multi-exposure image fusion, the exposure quality is a crucial factor. In this work, we adopt the simple threshold-based quality measure introduced in [66] to generate the weight map for exposure quality. The focus map obtained from the CNN model still acts as the weight map for contrast extraction. These two kinds of weight maps are multiplied to generate the final weight map as presented in [67]. Finally, the same multi-scale fusion scheme based on LP and GP as mentioned above is employed to obtain the fused image.

Three fusion examples are shown in Fig. 14, where the two source images are listed in the first two columns, and the third column shows the fused image using the presented CNN-based methods. The multi-modal image fusion algorithm is used in both visible-infrared and medical examples. It can be seen that the fused image well preserves important information in the source

images. The fusion quality of multi-exposure images is also relatively high, as the fused image extracts most spatial details without introducing undesirable artifacts.

Considering the presented methods in this subsection together with the CNN-based multi-focus image fusion method in Section 3, one can find that the techniques applied by different image fusion issues are not the same. However, they share the CNN mapping process starting from source images to the focus map, which is the common core task of various image fusion issues as this mapping process simultaneously involves activity level measurement and comparison (namely, fusion rule). The subsequent techniques applied to the focus map could be selected or designed according to the characteristics of a specific fusion task. This is a reasonable way to study this topic from our perspective as we believe conventional techniques in related fields are still of high value and should not be discarded. In this work, we just employ some popular techniques for either multi-focus image fusion or other-type fusion issues, so further studies following this route could be performed in the future.

4.4. Future directions

It is worthwhile to notice that this paper just provides a preliminary attempt aiming to exhibit the great potential of CNN for image fusion. In the future, more in-depth work could be carried out on this topic from the following three aspects.

1) *Design of network architecture*. Further investigation on network design for image fusion is a meaningful task. The siamese network is employed for image fusion in this work, but the pseudo-siamese and 2-channel networks are also applicable. In Section 4.2.4, we show that the performance of the fusion method only decreases a little when removing a fully-connected layer, so the impact of network depth on the fusion performance is also worth studying. Another interesting issue is the size of training patches, which is coupled with network architecture. Furthermore, the network designed in this paper is essentially a classification network, while one label (0 or 1) is set to a training patch pair. This may lead to inaccuracy around the boundaries between focused and defocused regions. Another CNN-based approach is designing a real end-to-end network for image fusion that the output could be the final decision map or even the fused image. The main difficulty of this idea is that there is theoretically no ground-truth fused images for training. But for multi-focus image fusion, this difficulty may be overcome since it is possible to manually create ground-truth results [26], although the workload is huge.
2) *Development of more complicated fusion schemes*. In this work, the patch size is set to 16 mainly to improve the fusion quality of boundary regions, but it will sacrifice some accuracy of focus/defocus classification. One possible improved approach is to train several CNN models with different input size and using them jointly to pursue a better performance. As mentioned above, the image processing techniques performed on the focus map in this work are relatively simple, so one can also develop more elaborate techniques or complicated strategies to improve the fusion quality.
3) *Extension to other-type image fusion tasks*. In our algorithm, the focus map obtained from the learned CNN model contains the information about spatial details, which is of great importance to other-type image fusion tasks, such as visible-infrared image fusion, multi-modal medical image fusion and multi-exposure image fusion. In Section 4.3, we have briefly exhibited the potential of the learned CNN model for these fusion tasks. Obviously, there still exists plenty of room in this direction by designing more effective fusion schemes.

5. Conclusion

This paper mainly presents a new multi-focus image fusion method based on a deep convolutional neural network. The main novelty of our method is learning a CNN model to achieve a direct mapping between source images and the focus map. Based on this idea, the activity level measurement and fusion rule can be jointly generated by learning the CNN model, which can overcome the difficulty faced by the existing fusion methods. The main contribution of this paper could be summarized into the following four points: 1) We introduce CNNs into the field of image fusion. The feasibility and superiority of CNNs used for image fusion are discussed. It is the first time that CNNs are employed for an image fusion task to the best of our knowledge. 2) We propose a multi-focus image fusion method based on a CNN model. Experimental results demonstrate the proposed method can achieve state-of-the-art results in terms of visual quality and objective assessment. 3) We exhibit the potential of the learned CNN model for other-type image fusion issues. 4) We put forward some suggestions on the future study of CNN-based image fusion. We believe CNNs are capable of opening a new research approach in the field of image fusion. The MATLAB implementation of the proposed multi-focus image fusion method is available online at http://home.ustc.edu.cn/~liuyu1.

Acknowledgements

The authors would like to thank the editors and anonymous reviewers for their insightful comments and constructive suggestions. The authors sincerely thank Baocai Yin from IFLYTEK for his meaningful discussions. The authors would also like to express their thanks to Xudong Kang, Zhiqiang Zhou, Yu Zhang, Mansour Nejati and Zheng Liu for providing their codes and source images. This work was supported by the National Natural Science Foundation of China (Grants 81571760 and 61501164), the Research and Development Foundation of Hefei Research Institutes (Grant IMICZ2015111), and the Fundamental Research Funds for the Central Universities (Grants JZ2016HGPA0731 and JZ2016HGBZ1025).

References

[1] T. Stathaki, Image Fusion: Algorithms and Applications, Academic Press, 2008.
[2] P. Burt, E. Adelson, The Laplacian pyramid as a compact image code, IEEE Trans. Commun. 31 (4) (1983) 532–540.
[3] A. Toet, A morphological pyramidal image decomposition, Pattern Recognit. Lett. 9 (4) (1989) 255–261.
[4] H. Li, B. Manjunath, S. Mitra, Multisensor image fusion using the wavelet transform, Graphical Models Image Process. 57 (3) (1995) 235–245.
[5] J. Lewis, R. Oí Callaghan, S. Nikolov, D. Bull, N. Canagarajah, Pixel- and region-based image fusion with complex wavelets, Inf. Fusion 8 (2) (2007) 119–130.
[6] Q. Zhang, B. Guo, Multifocus image fusion using the nonsubsampled contourlet transform, Signal Process. 89 (7) (2009) 1334–1346.
[7] G. Piella, A general framework for multiresolution image fusion: from pixels to regions, Inf. Fusion 4 (4) (2003) 259–280.
[8] X. Qu, J. Yan, H. Xiao, Z. Zhu, Image fusion algorithm based on spatial frequency-motivated pulse coupled neural networks in nonsubsampled contourlet transform domain, Acta Autom. Sin. 34 (12) (2008) 1508–1514.
[9] J. Tian, L. Chen, Adaptive multi-focus image fusion using a wavelet-based statistical sharpness measure, Signal Process. 92 (9) (2012) 2137–2146.
[10] X. Li, H. Li, Z. Yu, Y. Kong, Multifocus image fusion scheme based on the multiscale curvature in nonsubsampled contourlet transform domain, Opt. Eng. 54 (2015) 073115.
[11] Y. Liu, S. Liu, Z. Wang, A general framework for image fusion based on multi-scale transform and sparse representation, Inf. Fusion 24 (1) (2015) 147–164.
[12] N. Mitianoudis, T. Stathaki, Pixel-based and region-based image fusion schemes using ica bases, Inf. Fusion 8 (2) (2007) 131–142.
[13] B. Yang, S. Li, Multifocus image fusion and restoration with sparse representation, IEEE Trans. Instrum. Meas. 59 (4) (2010) 884–892.
[14] J. Liang, Y. He, D. Liu, X. Zeng, Image fusion using higher order singular value decomposition, IEEE Trans. Image Process. 21 (5) (2012) 2898–2909.
[15] Y. Jiang, M. Wang, Image fusion with morphological component analysis, Inf. Fusion 18 (1) (2014) 107–118.
[16] Z. Liu, Y. Chai, H. Yin, J. Zhou, Z. Zhu, A novel multi-focus image fusion approach based on image decomposition, Inf. Fusion 35 (2017) 102–116.
[17] W. Huang, Z. Jing, Evaluation of focus measures in multi-focus image fusion, Pattern Recognit. Lett. 28 (4) (2007) 493–500.

[18] S. Li, J. Kwok, Y. Wang, Combination of images with diverse focuses using the spatial frequency, Inf. Fusion 2 (3) (2001) 169–176.
[19] S. Li, J. Kwok, Y. Wang, Multifocus image fusion using artificial neural networks, Pattern Recognit. Lett. 23 (8) (2002) 985–997.
[20] V. Aslantas, R. Kurban, Fusion of multi-focus images using differential evolution algorithm, Expert Syst. Appl. 37 (12) (2010) 8861–8870.
[21] I. De, B. Chanda, Multi-focus image fusion using a morphology-based focus measure in a quad-tree structure, Inf. Fusion 14 (2) (2013) 136–146.
[22] X. Bai, Y. Zhang, F. Zhou, B. Xue, Quadtree-based multi-focus image fusion using a weighted focus-measure, Inf. Fusion 22 (1) (2015) 105–118.
[23] M. Li, W. Cai, Z. Tan, A region-based multi-sensor image fusion scheme using pulse-coupled neural network, Pattern Recognit. Lett. 27 (16) (2006) 1948–1956.
[24] S. Li, B. Yang, Multifocus image fusion using region segmentation and spatial frequency, Image Vis. Comput. 26 (7) (2008) 971–979.
[25] S. Li, X. Kang, J. Hu, Image fusion with guided filtering, IEEE Trans. Image Process. 22 (7) (2013) 2864–2875.
[26] S. Li, X. Kang, J. Hu, B. Y, Image matting for fusion of multi-focus images in dynamic scenes, Inf. Fusion 14 (2) (2013) 147–162.
[27] Z. Zhou, S. Li, B. Wang, Multi-scale weighted gradient-based fusion for multi–focus images, Inf. Fusion 20 (1) (2014) 60–72.
[28] D. Guo, J. Yan, X. Qu, High quality multi-focus image fusion using self-similarity and depth information, Opt. Commun. 338 (1) (2015) 138–144.
[29] Y. Liu, S. Liu, Z. Wang, Multi-focus image fusion with dense sift, Inf. Fusion 23 (1) (2015) 139–155.
[30] M. Nejati, S. Samavi, S. Shirani, Multi-focus image fusion using dictionary-based sparse representation, Inf. Fusion 25 (1) (2015) 72–84.
[31] Y. Zhang, X. Bai, T. Wang, Boundary finding based multi-focus image fusion through multi-scale morphological focus-measure, Inf. Fusion 35 (2017) 81–101.
[32] S. Li, X. Kang, L. Fang, J. Hu, H. Yin, Pixel-level image fusion: a survey of the state of the art, Inf. Fusion 33 (2017) 100–112.
[33] G. Gao, L. Xu, D. Feng, Multi-focus image fusion based on non-subsampled shearlet transform, IET Image Process. 7 (6) (2013) 633–639.
[34] H. Zhao, Z. Shang, Y. Tang, B. Fang, Multi-focus image fusion based on the neighbor distance, Pattern Recognit. 46 (3) (2013) 1002–1011.
[35] B. Yang, S. Li, Pixel-level image fusion with simultaneous orthogonal matching pursuit, Inf. Fusion 13 (1) (2012) 10–19.
[36] N. Yu, T. Qiu, F. Bi, A. Wang, Image features extraction and fusion based on joint sparse representation, IEEE J. Sel. Top Signal Process. 5 (5) (2011) 1074–1082.
[37] Y. Liu, Z. Wang, Simultaneous image fusion and denosing with adaptive sparse representation, IET Image Process. 9 (5) (2015) 347–357.
[38] Y. LeCun, L. Bottou, Y. Bengio, P. Haffner, Gradient-based leaning applied to document recognition, Proc. IEEE 86 (11) (1998) 2278–2324.
[39] P. Sermanet, D. Eigen, X. Zhang, M. Mathieu, R. Fergus, Y. L, Overfeat: integrated recognition, localizaton and detection using convolutional networks, arXiv 1312.6299v4 (2014) 1–16.
[40] https://en.wikipedia.org/wiki/Deep_learning.
[41] V. Nair, G. Hinton, Rectified linear units improve restricted boltzmann machines, in: Proceedings of 27th International Conference on Machine Learning, 2010, pp. 807–814.
[42] A. Krizhevsky, I. Sutskever, G. Hinton, Imagenet classification with deep convolutional neural networks, in: Advances in Neural Information Processing Systems, 2012, pp. 1097–1105.
[43] S. Farfade, M. Saberian, L. Li, Multi-view face detection using deep convolutional neural networks, in: Proceedings of the 5th ACM on International Conference on Multimedia Retrieval, 2015, pp. 643–650.
[44] Y. Sun, X. Wang, X. Tang, Deep learning face representation from predicting 10,000 classes, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2014, pp. 1891–1898.
[45] J. Long, E. Shelhamer, T. Darrell, Fully convolutional networks for semantic segmentation, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015, pp. 3431–3440.
[46] C. Dong, C. Loy, K. He, X. Tang, Image super-resolution using deep convolutional networks, IEEE Trans. Pattern Anal. Mach. Intell. 38 (2) (2016) 295–307.
[47] S. Zagoruyko, N. Komodakis, Learning to compare image patches via convolutional neural networks, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015, pp. 4353–4361.
[48] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, T. Darrell, Caffe: convolutional architecture for fast feature embedding, in: Proceedings of the ACM International Conference on Multimedia, 2014, pp. 675–678.
[49] http://www.vlfeat.org/matconvnet/.
[50] K. He, J. Sun, X. Tang, Guided image filtering, IEEE Trans. Pattern Anal. Mach. Intell. 35 (6) (2013) 1397–1409.
[51] http://www.image-net.org/.
[52] X. Glorot, Y. Bengio, Understanding the difficulty of training deep feedforward neural networks, in: International Conference on Artificial Intelligence and Statistics, 2010.
[53] http://mansournejati.ece.iut.ac.ir/content/lytro-multi-focus-dataset.
[54] S. Li, B. Yang, J. Hu, Performance comparison of different multi-resolution transforms for image fusion, Inf. Fusion 12 (2) (2011) 74–84.
[55] http://home.ustc.edu.cn/~liuyu1.
[56] http://xudongkang.weebly.com/index.html.
[57] https://github.com/lsauto/MWGF-Fusion.
[58] V. Petrovic, V. Dimitrijevic, Focused pooling for image fusion evalution, Inf. Fusion 22 (1) (2015) 119–126.
[59] Z. Liu, E. Blasch, Z. Xue, J. Zhao, R. Laganiere, W. Wu, Objective assessment of multiresolution image fusion algorithms for context enhancement in night vision: a comparative study, IEEE Trans. Pattern Anal. Mach. Intell. 34 (1) (2012) 94–109.
[60] M. Hossny, S. Nahavandi, D. Creighton, Comments on information measure for performance of image fusion, Electron. Lett. 44 (18) (2008) 1066–1067.
[61] C.S. Xydeas, V.S. Petrovic, Objective image fusion performance measure, Electron. Lett. 36 (4) (2000) 308–309.
[62] C. Yang, J. Zhang, X. Wang, X. Liu, A novel similarity based quality metric for image fusion, Inf. Fusion 9 (2) (2008) 156–160.
[63] Y. Chen, R. Blum, A new automated quality assessment algorithm for image fusion, Image Vis. Comput. 27 (10) (2009) 1421–1432.
[64] Z. Wang, A. Bovik, H. Sheikh, E. Simoncelli, Image quality assessment: from error visibility to structural similarity, IEEE Trans. Image Process. 13 (4) (2004) 600–612.
[65] M. Hossny, S. Nahavandi, D. Creighton, A. Bhatti, Image fusion performance metric based on mutual information and entropy driven quadtree decomposition, Electron. Lett. 46 (18) (2010) 1266–1268.
[66] W. Zhang, W.-K. Cham, Gradient-directed multiexposure composition, IEEE Trans. Image Process. 21 (4) (2012) 2318–2323.
[67] T. Mertens, J. Kautz, F.V. Reeth, Exposure fusion, in: Proceedings of Pacific Graphics, 2007, pp. 382–390.