Journal of Ambient Intelligence and Humanized Computing (2023) 14:14795–14805
https://doi.org/10.1007/s12652-018-1036-4

ORIGINAL RESEARCH

Image denoising via deep network based on edge enhancement

Xi Chen[1] · Shu Zhan[1] · Dong Ji[1] · Liangfeng Xu[1] · Congzhong Wu[1] · Xiaohong Li[1]

Received: 4 February 2018 / Accepted: 19 August 2018 / Published online: 18 September 2018

Abstract
Existing methods for image denoising mainly focused on noise and visual artifacts too much but rarely mentioned the loss of edge information. In this paper, we propose a deep denoising network based on the residual learning and perceptual loss to generate high-quality denoised results. Inspired by the deep residual network, two new strategies are used to modify the original structure, which can improve the learning process by compressing the mapping range. At first, the high-frequency layer of noisy image is used as the input by removing background information. The secondly, a residual mapping is trained to predict the difference between clean and noisy images as output instead of the final denoised image. Furtherly improve the denoised result, a joint loss function is defined as the weighted sum of pixel-to-pixel Euclidean loss and perceptual loss. A well-trained convolutional neural network is connected to learn the semantic information we would like to measure in our perceptual loss. It encourages the train process to learn similar feature representations rather than match each low-level pixel, which can guide front denoising network to reconstruct more edges and details. As opposed to the standard denoising models that concentrate on one specific noise level, our single model can deal with the noise of unknown levels (i.e., blind denoising). The experiments show that our proposed network achieves superior performances and recovers majority of missing details from low-quality observations.

Keywords Image denoising · Residual learning · Joint perceptual loss · Edge enhancement · Hierarchical mode

1 Introduction

Image denoising is a classical image reconstruction problem in low-level computer vision that estimates the latent clean image from a noisy one. In reality, noise caused by imaging equipment and external environment during the process of digitization and transmission affects digital images. As a result, denoising is an essential practice to improve image quality. It gives a picture a better visual performance as demonstrated by the features of the denoised picture. It is vital for various applications ranging from camera imaging, medical imaging to video surveillance imaging as well as satellite remote sensing image processing among others. The linear form of the degeneration model of images is

$$x = y + n \tag{1}$$

where y is a clean image, x is the noisy target to be restored, and n is additive noise. Similar to image separation (Zhang 2016), image denoising can be simply viewed as the problem of separating two components from a noisy image. During the acquisition of images from imaging system, the actual system suffers from a variety of noise disturbances. The most common of these are Gaussian distributed thermal noise generated by electromagnetic interference during imaging. Therefore, this paper aims at solving problem of recovering a clean image from a noisy observation that assumes additive white Gaussian noise with a standard deviation σ.

Over the past few decades, the image prior played a central role in highly ill-posed problem (when the likelihood is known), thus various methods have been proposed for exploiting image priors, including nonlocal self-similarity (NSS) models (Buades et al. 2005; Dabov et al. 2007; Gu et al. 2014), regularization models (Rudin et al. 1992; Osher et al. 2011; Ono and Yamada 2016), sparse models (Elad and Aharon 2006; Zha et al. 2017), Markov random field (MRF) models (Lan et al. 2006). Despite the good denoising results that such methods can achieve, traditional models usually involve manual selection of parameters, and because

✉ Shu Zhan
shu_zhan@hfut.edu.cn

[1] School of Computer and Information, Hefei University of Technology, Hefei 230601, China

of complex optimization problems, these methods require huge amounts of time and computing costs. Instead of that, since the superior performance of deep learning model had been shown in ILSVRC2012 in image classification (Krizhevsky et al. 2012), more and more researchers have made efforts to propose methods of learning an end-to-end model which can directly map input to output (Zhu et al. 2017). Since the development of deep learning method and convolutional neural network tackling low-level vision tasks, e.g., image super-resolution, deblurring, inpainting. For denoising (Jain and Seung 2008) was the first to use convolutional neural networks. And through comparative experiments, it has been showed that CNNs achieve promising performance that is even better than MRF methods (Lan et al. 2006). After that, more deep learning methods (Burger et al. 2012; Xie et al. 2012; Chen and Pock 2017) were proposed for denoising, and they also had a promising quality to compete with BM3D.

Recently, due to the powerful learning ability, a very deep convolutional neural network has widely been used in image restoration. Inspired by the ResNet, residual learning has gradually been applied to perform various tasks, e.g., image super-resolution (Kim et al. 2016a, b), deblurring (Nah et al. 2017), denoising (Zhang et al. 2017). Consequently, network structures have become deeper, good at learning, and have better features for higher quality results.

Despite the pleasant performances in removing the noise, most of the denoising methods typically suffer from a common drawback. They generate blurred image without sharp edge and fine detail as shown in Fig. 1d, e. Since details such as sharp edges and texture information are vital for image denoising, it is desirable to keep more features from the input as possible. To address this challenging problem,

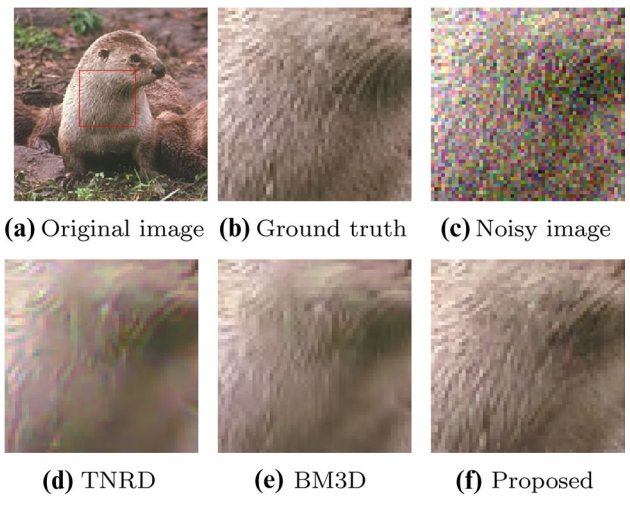


Fig. 1 Denoising results for an image corrupted by the Gaussian noise with $\sigma = 25$. Our result preserves more details, while other results have artifacts or blurred edges

we develop an end-to-end residual architecture based on detail enhancement to eliminate Gaussian noise while keeping more precise visual results. The experiments show that, compared with others, our proposed model can generates more clean output and reconstructs more details. As shown in Fig. 1f, the restored pictures are more like the original ones, especially regarding the animal hair.

1.1 Our contribution

Different from other direct denoising network, we introduce the residual units to solve the problem of gradient vanishing in deep structure. In addition to this, two methods are proposed to simplify the training difficulty. One is to remove irrelevant background information from the noisy input, and the other is learning residual mapping to generate the noise in a image. Relatively direct network, this can make the training process easier.

We modified the traditional denoising loss function. Given a well-trained convolutional neural network, we instead to learn the perceptual differences of extracted features, rather than just matching low-level pixel information. Different from the lost of detail by using normal per-pixel loss solely, a new joint loss is recommended to get more edge and detail information.

2 Related work

In the past few years, research focusing on this area shifted to how to make the best use of image priors. As a result, the nonlocal self-similar patch matching (NLM) (Buades et al. 2005) and module matching with collaborative filtering (BM3D) (Dabov et al. 2007) have been two outstanding baselines for image denoising for almost a decade now.

Following the success of AlexNet in ImageNet, the era of deep learning for high-level vision started and as more researchers tried to apply deep learning to image denoising tasks. In an earlier attempt, Jain and Seung (2008) proposed a simple CNN to recover a clean natural image from a noisy observation and created the precedent of deep learning us in denoising. Burger et al. (2012) proposed a multiple layer perceptron and Xie et al. (2012) introduced an auto-encoders network. These methods proved they could get better results. Chen and Pock (2017) proposed a trainable nonlinear reaction diffusion (TNRD) model, which has a feed-forward deep network adapting field-of-experts by doing a fixed number of gradient descent steps. Obviously, TNRD got better results compared to the more classical methods. However, priors imposing need to rely on parameters settings and extensive fine-tuning.

Recently, more complex structures, like GoogleNet and ResNet (He et al. 2015), have revealed that the network

depth is vital. Consequently, very deep CNNs have become famous for image restoration. The first deep network with 20 convolutional layers was proposed for SISR. And it significantly enhanced the ability to learn features (Kim et al. 2016a). For denoising, Mao et al. (2016) proposed a very deep auto-encoder network, each pair of encoding and decoding made the feature mappings into different scales. Inspired by residual strategy, Zhang et al. (2017) introduced batch normalization into a DnCNN model to speed up the training.

3 Proposed method

This section describes the proposed model architecture and the training loss function. Our model consists of two parts. The first part is a denoising network, which is based on an optimized residual network to comply the noise eliminating step. The other has a loss network based on well-trained CNN to define the perceptual loss.

3.1 Denoising network

Figure 2 indicates the structure of the denoising model that includes the residual network, high-frequency layer decomposition, and global skipping connection. With in them, the convolution layers mainly comprise many residual units, which realize end-to-end mapping learning of images. The other two parts are used to optimize the residual network. They assist in simplifying the training process by reducing the solution space. As shown below, high-frequency layer and residual of noisy image are used as the input and output to the middle convolutional network instead of original images.

3.1.1 Residual network

The central part of the denoising model is residual network. The deep architecture is shown in Fig. 3, and there are three types of modules with different functions. The embedding module is equivalent to the current patch generation and feature extraction. The residual module, with a chain of residual units, is used to learn end-to-end mapping. And the reconstruction module is the post-processing step that generates the denoised output image.

Embedding module: Feature extraction is the most crucial step in the entire model. The first embedding module maps the input from image space to feature space. We use a 3×3 convolution layer and a ReLU activation function as the embedding part, 64 filters are utilized to generate 64 feature maps.

Residual module: Inspired by ResNet, we introduce identity mappings in our network. During the forward and back-propagation phases of training, the signal passes directly from one unit to another, resulting in the reduction of the possibility of gradient vanishing.

The design consists of a chain of residual units. Each unit is divided into two branches: a residual and an identity mapping. The residual branch contains two layers which aim to learn convolution filters to extract features for denoising. Besides, the identity mapping allows gradients to transfer directly in propagation. Each unit can expressed as,

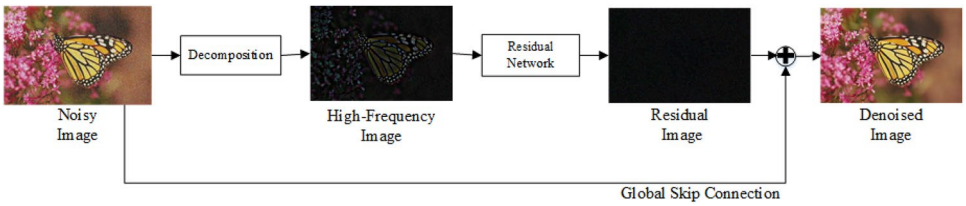


Fig. 2 The proposed denoising network architecture, which consists of high frequency decomposition, residual network, and a global skip connection. The middle images show a better visualization of the process

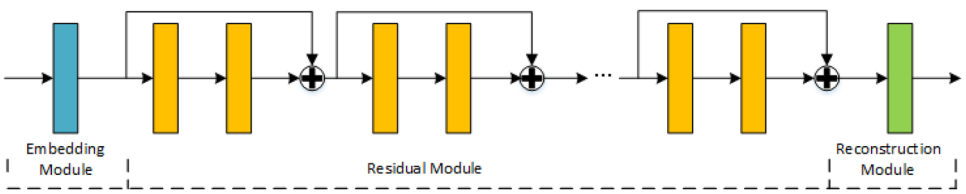


Fig. 3 The overall structure of the residual network, where the first mode is the embedding module, the middle modes make up the residual module, the last mode is the reconstruction module. The residual module consists of a chain of residual units, and each unit includes two convolution layers and an identity mapping

$$R^u = F(R^{u-1}) + R^{u-1} \tag{2}$$

where R represents each unit, u = 1,2, U, U is the number of residual units, F is the residual function. Instead of directly using the popular ResNet, we slightly modify its structure. Inspired by He et al. (2016) where pre-activation has offered to achieve the better performance for classification, we remove the rectified linear unit before every convolution layer. Pre-activation improves the convergence rate during training, while the identity function retains the range of gradient. Moreover, the resulting network is more generic than the one using after-activation.

Reconstruction module: It is necessary to transform the multi-channel of final feature maps back into the original image space (1 or 3-channel). Therefore, we define a 3×3 convolutional layer as the reconstruction part to produce an accurate image.

3.1.2 Reducing mapping range

For image restoration problems such as super-resolution and denoising, image details are of great importance. Aware of that, like most restoration models, we remove the layers for dimension reduction which smooth the features, such like pooling. As a result, the solution space covers all pixel values, which increases the difficulty of learning the regression function well. Accordingly, we propose two methods to make the learning process easier through reducing the mapping range of residual network. The first method is erasing the background information of input, and the other is learning the residual of noise as the output.

High frequency decomposition: The denoising task mainly deals with the noise in an image. If other unrelated information in the input can be removed, the solution space of network will be greatly compressed. Refer to the traditional theory of space domain, a raw image X is decomposed as the sum of a low-frequency layer and a high-frequency layer,

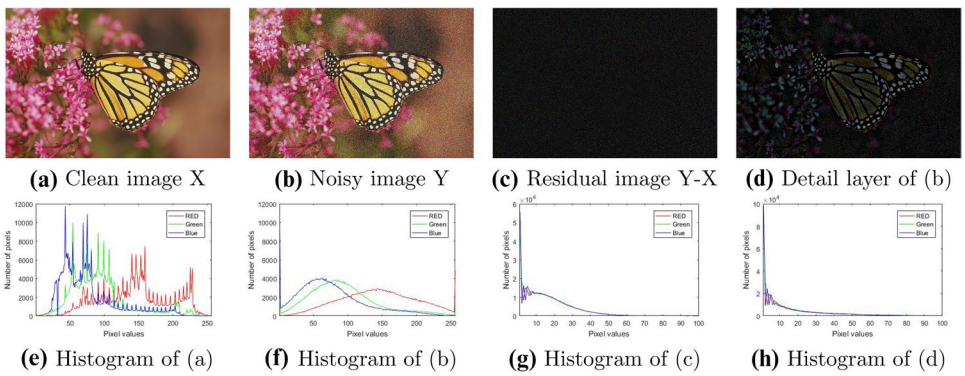
$$X = X_{low} + X_{high} \tag{3}$$

The low-frequency layer is easily obtained by low-pass filters (He et al. 2013), and all the base backgrounds are in this layer. After removing the low-frequency layer from an image, only variable pixels, such as sharp edges and noisy points, remain in the high-frequency layer. In Fig. 4c, d, we respectively display the base layers of clean image and the corresponding noisy one. We subjectively believe that two pictures contain the same background information in visual. For a more objective contrast, the residual image of two base layers is shown in Fig. 4e, and it has little information. This implies that most of noise really remains in the high-frequency layer. Accordingly, the high-frequency layer can be utilized instead of the original image as the input to residual network. This hierarchical mode is usually used for video coding and processing (Zeng et al. 2010, 2011, 2014; Wang et al. 2011), and a few works have applied it in deep learning framework of denoising.

As shown in Fig. 5f, h, the high-frequency layer is more sparse than the original noisy image since most of its pixels are close to zero. This proves that erasing the background information achieves range compression efficiently. With such strategy, a deeper network can make it easier to train, so that this sparsity further improves the denoising performance. Moreover the low-frequency information do not need to put into the network for processing, which makes the background information free from being smooth overly.

(a) Clean image X **(b)** Noisy image Y **(c)** Base layer X_{low} **(d)** Base layer Y_{low} **(e)** $Y_{low} - X_{low}$

Fig. 4 Example low-frequency layer of a image. Few difference between **c** and **d** means noise almost remain in the high-frequency



(a) Clean image X **(b)** Noisy image Y **(c)** Residual image Y-X **(d)** Detail layer of (b)

(e) Histogram of (a) **(f)** Histogram of (b) **(g)** Histogram of (c) **(h)** Histogram of (d)

Fig. 5 Range reduction and sparsity of the residual image and high-frequency layer. Different from the even distribution of original clean and noisy images, both residual and high-frequency layer have the narrow range

Residual mapping: Similar to reducing the sparsity of input, we replace the output with residual which is the difference between noisy image and ground truth. As shown in Fig. 2, we learn the noise residual mapping $F(x_{high}) = n$ first, and then estimate the final denoised image y via a global skip connection to remove the residual n from noisy image x. The mathematical expression for this is,

$$y = x - F(x_{high}) \tag{4}$$

By comparing Fig. 5e, g, it can be seen that the solution space is indeed reduced. As opposed to the evenly distributed histogram of clean image X, the histogram of residual Y–X shows a more concentrated distribution, which has a noticeable space reduction in pixel values. This indicates using residual as the output of parameter layers helps to compress mapping range. In addition, like normal identity mapping, the global skip connection can also put gradients directly flow to the front convolutional layer to deal with the vanishing gradient problem.

3.2 Joint loss

Traditionally, learning based image restoration works used a per-pixel loss between ground truth and restored image as the optimization target, which can achieve excellent quantitative scores. However, in recent researches, minimizing pixel-wise errors depending only on low-level pixels has proved that it may cause the loss of details and make the results smooth (Johnson et al. 2016; Ledig et al. 2016). On the other hand, the perceptual loss function has shown it can generate high-quality images with a better visual performance by capturing the difference of high-level feature representations, but sometimes it fails to preserve color and local spatial information. To combine both benefits, we propose a new joint loss function consisting of a normal pixel-to-pixel loss and a perceptual loss together with appropriate weights (Fig. 6). It is defined as follows,

$$L_{Joint} = L_{mse} + \lambda L_{Per} \tag{5}$$

where L_{mse} is pixel-to-pixel MSE loss, L_{Per} represents perceptual loss.

3.2.1 MSE loss

Same as the existing network architectures for denoising, the MSE loss is used and calculated as,

$$L_{mse} = \frac{1}{CWH}\left\|F\left(x_{high}\right) - (x - y)\right\|^2 \tag{6}$$

where x is a noisy image, x_{high} is the high frequency of x, y is the corresponding ground truth, and C,W,H stand for the channel,width and height of the input image pair {x,y}, F means residual formulation to learn the residual.

Undoubtedly, minimizing pixel-wise errors between generated image and ground truth can bring particularly good quantitative scores of peak signal to noise ratio (PSNR). However, pre-pixel loss can not measure the perceptual and semantic differences between images. It can potentially output blurry images. Thus, we introduce a perceptual loss to address these shortcomings.

3.2.2 Perceptual loss

Inspired by recent works used in super resolution, we encourage the output image and target image have similar feature representations rather than only matching each low-level pixels of them. The key point is that the convolutional neural network pre-trained for image classification or semantic segmentation have already learned to encode semantic features, and these feature representations can be used directly in our perceptual loss. Thus, we employ a CNN with fixed weights as a loss network to extract the feature maps of denoised output and ground truth respectively from a certain middle layer. By comparing the feature values of both, the blurred regions of generated image

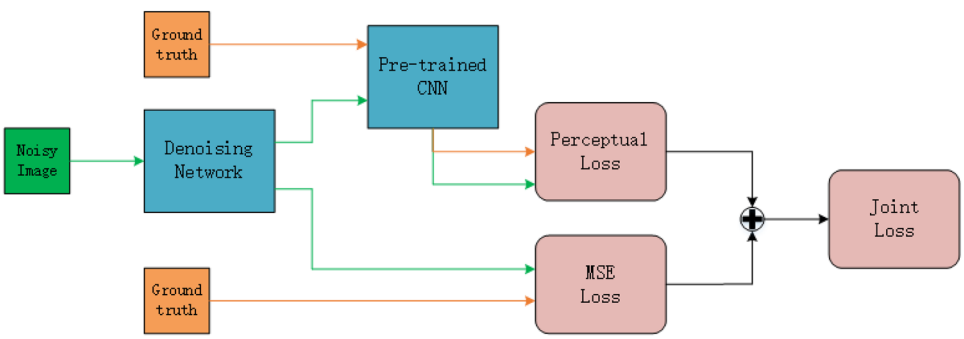


Fig. 6 Our proposed joint loss of cascaded network. First, a pair of images containing noisy image and ground truth are input to the denoising network to generate denoised image via MSE loss. Then we put the generated image and ground truth into the pre-trained CNN, and use the selected layers to extract feature maps of them to learn the differences of feature representations by perceptual loss

are reconstructed to be perceptually similar to the ground truth. In other words, the use of minimizing perceptual loss allows the transfer of texture feature knowledge from the loss network to the denoising network which can guide to reserve fine details of denoised results. Accordingly, the perceptual loss is also called the feature reconstruction loss.

In our experiments, we make use of a SegNet pre-trained for image semantic segmentation as the loss network (Badrinarayanan et al. 2017). There are a series of encoding convolution layers and symmetrical decoding layers in SegNet, where the upsampling and decoding layers are used to recover more spatial information lost due to downsampling. We define the perceptual loss function in the feature space, and measure it between the feature representations of ground truth and front networks output,

$$L_{Per} = \frac{1}{C_i W_i H_i} \left\| \varphi_i\left(x - F\left(x_{high}\right)\right) - \varphi_i(y) \right\|^2 \tag{7}$$

where, φ_i is the ReLU of ith convolution layer, $\varphi_i()$ stands for the feature map of shape $C_i \times W_i \times H_i$. x-F$\left(x_{high}\right)$ means the output generated by front denoising network, y means the ground truth. As for the feature extractor, we make φ_i be the fourth layer of encoder which mainly encodes texture and edge information (Johnson et al. 2016). In conclusion, we put the generated image and ground truth into a pre-trained CNN and extract their feature maps in a convolution layer respectively, then encourage them to have similar feature representations by minimizing the perceptual loss function, which guides the denoised results recover sharp edges and fine details.

4 Experiment

4.1 Training details

In our experiment, we compare the performance of our method with different baseline configurations. For different noise levels, we offer two ways to solve it. One is to train a specific net for each level separately. The other is blind denoising that a single net can deal with different levels noise. For training, besides the joint loss, we train a denoising network by using MSE singularly to facilitate comparison. Based on all the above, three models are used in contrast experiment.

MSE-S: the specific network with MSE loss
MSE-B: the blind network with MSE loss
Joint: the blind network with joint loss

All proposed networks are based on the public available deep library Caffe (Jia et al. 2014). All experiments are implemented in a single Nvidia GTX TITAN X GPU with 12G memory and CUDA edition is 7.5.

For parameter setting, we typically employ 64 traditional filters of size 3×3 in each convolution layer of training patch of size 40. In view of the better performance with deeper structure, we use 14 residual units to set the depth of denoising network to 26, and exploit SGD with momentum of 0.9 and a mini-batch size of 32. We start with a learning rate of 10^{-3}, and it is halved at every 1.2×10^5 iterations.

4.2 Datasets and baselines

Training data: Both our models, with the known or unknown noise level, learn from a training set same as Zhang et al. (2017) with 400 images of Berkeley segmentation dataset, which have already captured sufficient variability of natural images. Moreover, rather than training at the image-level, the original images are randomly cropped into 40×40 patches. To train network for known specific noise level, we generate the noisy images by adding Gaussian noise with standard deviations of $\sigma = 15, 25, 50$. Alternatively, we train a single blind network for the unknown noise range [1,50]. Similar to the gray images, color training dataset use the Berkeley segmentation dataset except CBSD68.

Testing data: For gray image model, we set up the test dataset from BSD68. These images are widely used for the evaluation of denoising methods. Note that there is no overlap between the training and evaluation. For color image model, we used two standard test sets: Set14 and CBSD68. Set14 are usually used to evaluate the advantages and disadvantages of traditional image processing methods. The BSD68 corresponds to a variety of natural scenes.

4.3 Results analysis

To verify the effectiveness of the proposed network, we show the quantitative and qualitative results of our method in comparison to state-of-the-art techniques, including BM3D, TNRD and DnCNN. Also, we compare the convergence of three deep networks with different structures: plain network, ResNet and our proposed network.

4.3.1 Quantitative results

We measure two evaluation indicators to evaluate the denoised results: peak signal-to-noise ratio (PSNR) and average gradient. PSNR is the most common criterion for image restoration tasks. A higher PSNR indicates that the restoration image is more like the ground truth by each low-level pixel. However, PSNR results can not be completely consistent with the visual quality sometimes.

Because PSNR only depends on the similarity with each low-level pixel exactly, and the feature reconstruction method learns perceptually similarity in feature space, which harms its PSNR. While PSNR results can no longer be used as the only standard for judging the quality of pictures, more new methods correlating with human visual system perception are being to define (Ni et al. 2016, 2017; Yang et al. 2016). In this experiment, we introduce average gradient as another criterion. It means the difference in the gray level near the edges or on both sides of the image. This rate of change can be used to indicate image clarity. Higher average gradient indicates more edge information. It just can be used to evaluate the ability to enhance the edge.

For grayscale network, the average PSNR and average gradient results of compered methods on the BSD68 dataset with noise level of 15, 25, 50 are shown in Table 1. Both MSE-S and MSE-B can achieve better results of PSNR. MSE-S outperforms all compered methods at each noise level. MSE-B for blind denoising can also be better than most traditional methods expect DnCNN at low levels because DnCNN is the model for specific level like MSE-S. As for the Joint model, its PSNR results are lower. Because the part of the feature reconstruction loss measures in feature space rather than image space, which harms its PSNR. Different from PSNR, the Joint model acquires the best result of average gradient. The results of the remaining methods are all lower than the Joint one. This shows that although the PSNR is not high, the denoised result of Joint has sufficient edge details and relatively clear quality.

Tables 2 and 3 are the collections of PSNR and average gradient values for color images denoising in Set14 and CBSD68 respectively. Same as the gray one, our proposed MSE-S model get the best PSNR results, MSE-B and DnCNN have almost the same effect. The Joint model also receives the highest average gradient and relatively low PSNR. That proves that training with pre-pixel MSE loss achieve better results with PSNR, but training with perceptual loss can keep more edge details after denoising.

4.3.2 Qualitative results

Visual comparisons with different methods are showed in this part. We select some representative sample pictures in different test sets. In Fig. 7, all the results using other state-of-the-art methods are overly smooth. However, the Joint model generates a clearer image with more edges, especially in the part of window. Figures 8, 9 and 10 illustrate the visual results of Set14 and CBSD68. One can see that the BM3D produces a smooth part without sharp details, TNRD tends to generate artifacts in the smooth region, DnCNN is likely to preserve sharp details but is not good enough. In contrast, the Joint with a perceptual loss not only recovers more lost details but also keep pleasant Visual effects. Especially in Fig. 9, the noisy image with strong texture gets a better denoised result by Joint, which illustrates our model can also work well for texture information.

To subjectively illustrate the effectiveness of the proposed joint loss, we compare the differences of each denoised images by various methods. We take the butterfly image in Set14 as an example, Fig. 11a, that shows the residual image of denoised images between Joint and MSE, Fig. 11b shows the residual between Joint and DnCNN. For the sake of comparison, we perform the same enhanced contrast processing

Table 1 PSNR/average gradient values averaged over 68 gray images in BSD68

σ	BM3D	TNRD	DnCNN	MSE-S	MSE-B	Joint
15	31.08/6.95	31.42/6.11	31.73/7.17	31.80/7.21	31.70/7.01	31.58/7.32
25	28.57/6.50	28.89/5.76	29.23/6.73	29.32/6.81	29.20/6.60	28.97/6.95
50	25.62/4.78	26.01/3.96	26.23/5.87	26.38/5.90	26.29/5.81	26.03/6.19

Table 2 PSNR/average gradient values averaged over 14 color images in Set14

σ	BM3D	TNRD	DnCNN	MSE-S	MSE-B	Joint
15	32.37/7.61	32.50/7.30	32.86/7.49	32.93/7.59	32.89/7.40	32.51/7.73
25	29.97/6.86	30.05/6.64	30.43/6.97	30.60/7.09	30.48/6.99	30.22/7.28
50	26.72/5.93	26.81/5.49	27.18/6.52	27.35/6.63	27.20/6.50	26.98/6.88

Table 3 PSNR/average gradient values averaged over 68 color images in CBSD68

σ	BM3D	TNRD	DnCNN	MSE-S	MSE-B	Joint
15	33.50/6.98	31.37/6.46	33.89/7.15	34.98/7.23	33.85/7.08	33.65/7.30
25	30.09/6.55	28.88/6.18	31.33/6.72	31.42/6.80	31.32/6.60	31.03/6.91
50	27.37/4.83	25.94/4.44	27.97/5.86	28.10/5.90	28.00/5.75	27.85/6.21

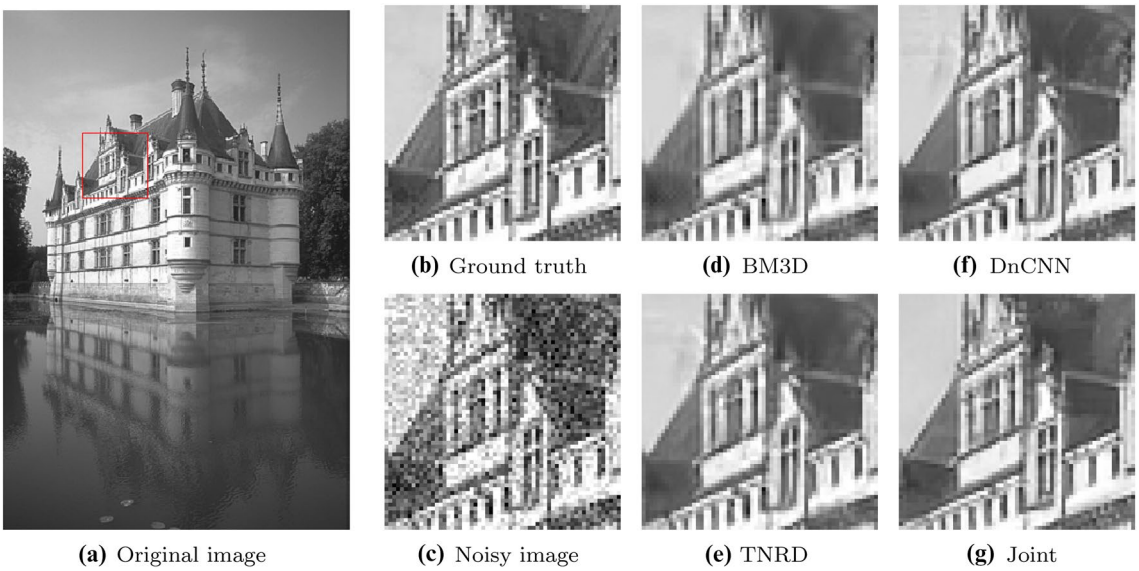


Fig. 7 Denoising performance comparison on sample gary image from BSD68

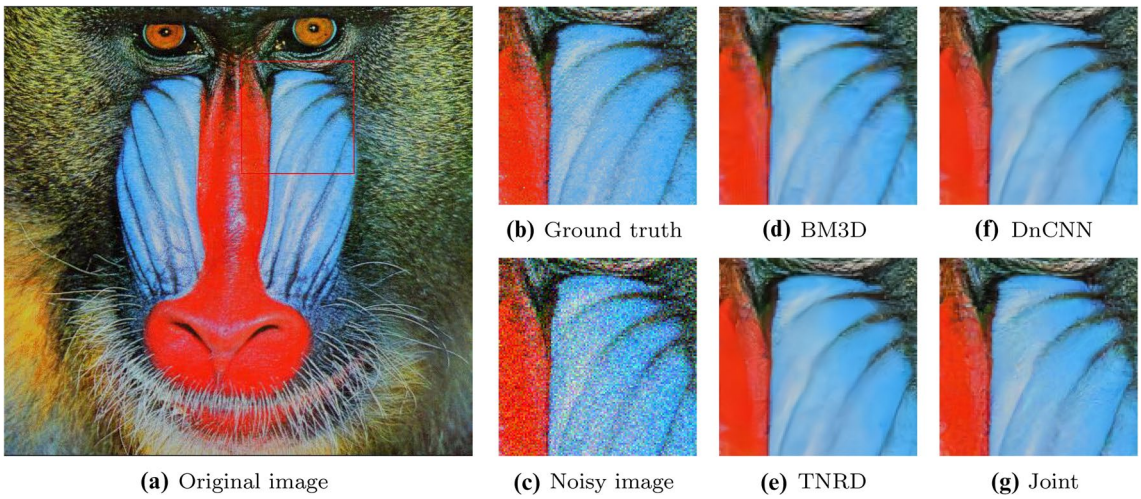


Fig. 8 Denoising performance comparison on sample color image from Set14

on two pictures. Form this experiment, it obvious that the Joint model contains more edge and texture information than MSE and DnCNN, especially the edges of butterfly wings and flowers. This demonstrates semantic information extracted from the loss network can guide denoising network to realize the edge enhancement.

To show the ability of blind denoising, we give an addition example as shown in Fig. 12. The input is composed of noisy parts with three levels, i.e., level 10 (the left), level 30 (the middle), level 50 (the right). In Fig. 12c, we see our blind model can generate satisfying restored output without artifacts even the input is corrupted by several levels of noise in different parts.

4.3.3 Convergence of different network structures

The convergence of the network is a very important indicator. The primary goal of training the network parameters is to make the network converge. In Fig. 13, we show the average per-image test loss of three types of deep networks: plain network, ResNet, proposed network with high frequency decomposition and residual mapping. It is obviously showed that the error of our proposed network has the fastest convergence rate and the best performance. The comparison with convergence illustrates the proposed high frequency decomposition and residual mapping really make the deep network train easier by reduce the mapping range.

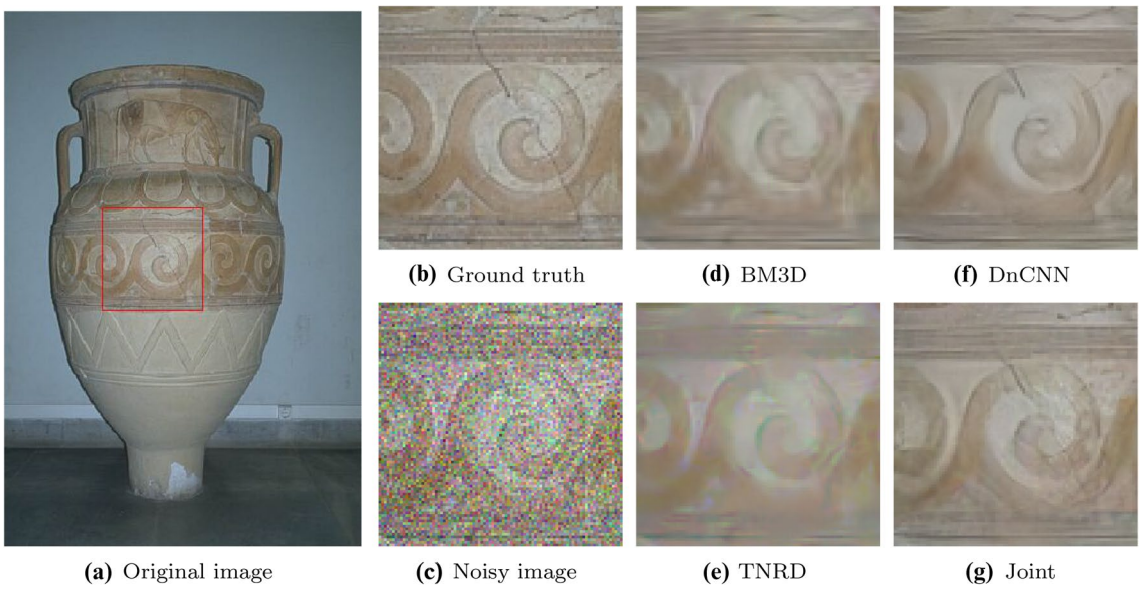


Fig. 9 Denoising performance comparison on sample color image with strong texture from CBSD68

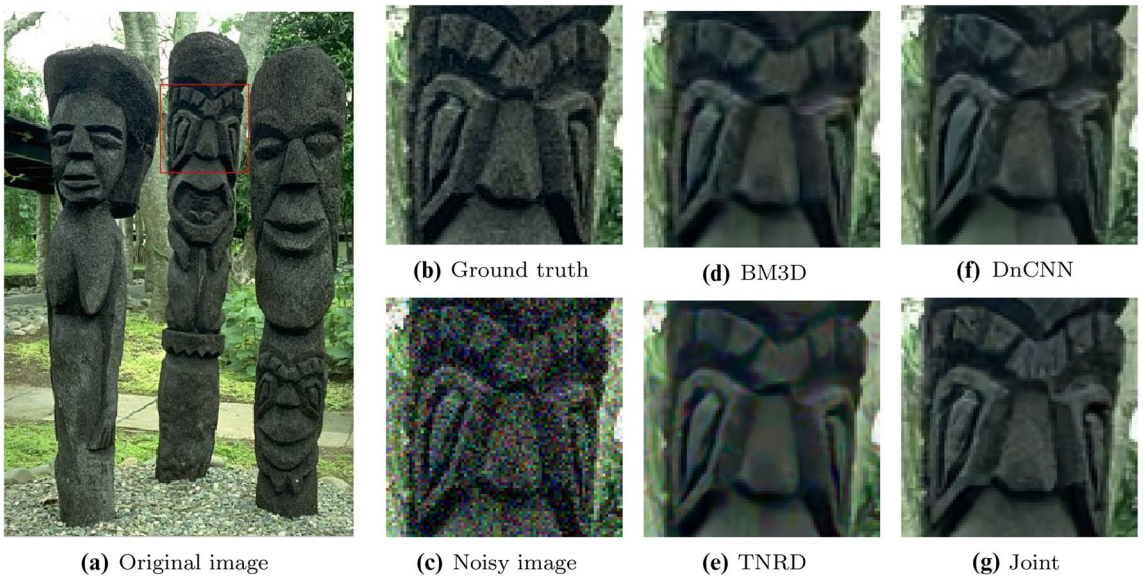
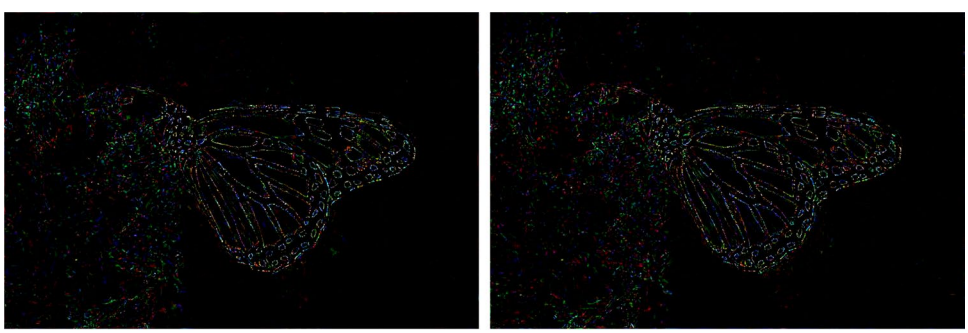
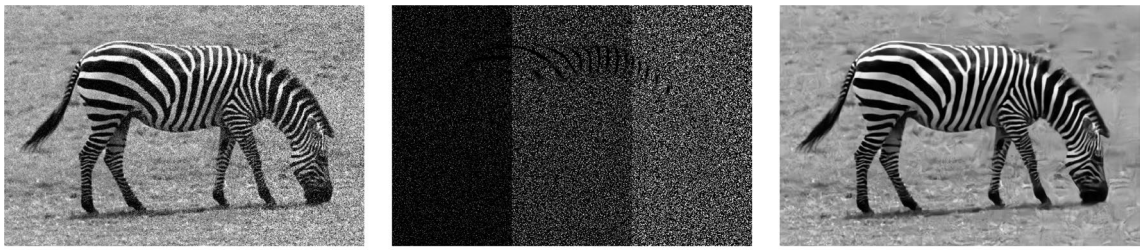


Fig. 10 Denoising performance comparison on sample color image from CBSD68

Fig. 11 The residual difference between denoised result of Joint model and other method





(a) Noisy image **(b)** Residual image **(c)** Denoised image

Fig. 12 An example showing the capacity of blind denoising. The input image is corrupted by the Gaussian noise with 10 (the left), 30 (the middle) and 50 (the right)

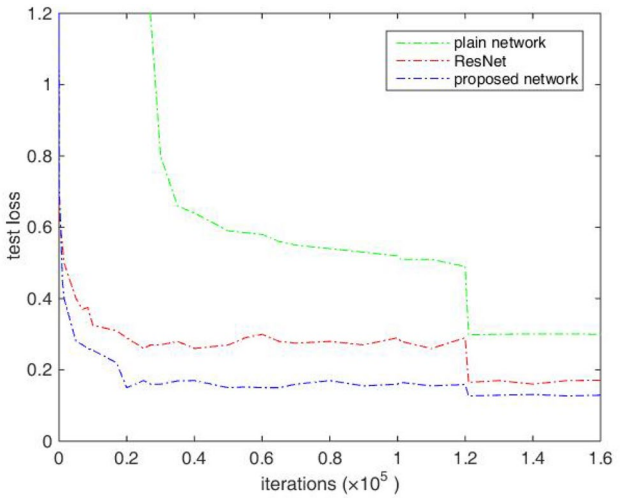


Fig. 13 Convergence of different network structures. The drops vertically 1.2×10^5 iterations are due to the learning rate division

5 Conclusion

In this paper, we have described a deep residual denoising network of 26 weight layers where perceptual loss is adopted to improve the denoising result with more edge and detail information. High-frequency layer decomposition and residual mapping are used to reduce the solution space in order to make the learning process easier. Different from the normal denoising model for only one specific noise level, our new model has the ability to deal with blind denoising problem with different unknown noise levels. For future work, we will explore to handle other kinds of noise, especially the complex real-world noise and consider a single comprehensive network for more image restoration tasks.

Acknowledgements This work was supported by National Nature Science Foundation of China Grant no. 61371156.The authors would like to thank the anonymous reviews for their helpful and constructive comments and suggestions regarding this manuscript.

References

Badrinarayanan V, Kendall A, Cipolla R (2017) Segnet: a deep convolutional encoder–decoder architecture for scene segmentation. IEEE Trans Pattern Anal Mach Intell 99:2481–2495

Buades A, Coll B, Morel JM (2005) A non-local algorithm for image denoising. In: Computer vision and pattern recognition, 2005. CVPR 2005. IEEE computer society conference on, vol 2, pp 60–65

Burger HC, Schuler CJ, Harmeling S (2012) Image denoising: can plain neural networks compete with BM3D? In: IEEE conference on computer vision and pattern recognition, pp 2392–2399

Chen Y, Pock T (2017) Trainable nonlinear reaction diffusion: a flexible framework for fast and effective image restoration. IEEE Trans Pattern Anal Mach Intell 39(6):1256–1272

Dabov K, Foi A, Katkovnik V, Egiazarian K (2007) Image denoising by sparse 3-D transform-domain collaborative filtering. IEEE Trans Image Process 16(8):2080–2095

Elad M, Aharon M (2006) Image denoising via sparse and redundant representations over learned dictionaries. IEEE Trans Image Process 15(12):3736–3745

Gu S, Zhang L, Zuo W, Feng X (2014) Weighted nuclear norm minimization with application to image denoising. In: IEEE conference on computer vision and pattern recognition, pp 2862–2869

He K, Sun J, Tang X (2013) Guided image filtering. IEEE Trans Pattern Anal Mach Intell 35(6):1397–1409

He K, Zhang X, Ren S, Sun J (2015) Deep residual learning for image recognition, pp 770–778

He K, Zhang X, Ren S, Sun J (2016) Identity mappings in deep residual networks, pp 630–645

Jain V, Seung HS (2008) Natural image denoising with convolutional networks. In: International conference on neural information processing systems, pp 769–776

Jia Y, Shelhamer E, Donahue J, Karayev S, Long J (2014) Caffe: convolutional architecture for fast feature embedding, pp 675–678

Johnson J, Alahi A, Li FF (2016) Perceptual losses for real-time style transfer and super-resolution. In: European conference on computer vision, pp 694–711

Kim J, Lee JK, Lee KM (2016a) Accurate image super-resolution using very deep convolutional networks. In: Computer vision and pattern recognition, pp 1646–1654

Kim J, Lee JK, Lee KM (2016b) Deeply-recursive convolutional network for image super-resolution. In: Computer vision and pattern recognition, pp 1637–1645

Krizhevsky A, Sutskever I, Hinton GE (2012) Imagenet classification with deep convolutional neural networks. In: International conference on neural information processing systems, pp 1097–1105

Lan X, Roth S, Huttenlocher D, Black MJ (2006) Efficient belief propagation with learned higher-order Markov random fields. In: European conference on computer vision, pp 269–282

Ledig C, Theis L, Huszar F, Caballero J, Cunningham A, Acosta A, Aitken A, Tejani A, Totz J, Wang Z (2016) Photo-realistic single image super-resolution using a generative adversarial network. In: IEEE conference on computer vision and pattern recognition, pp 105–114

Mao XJ, Shen C, Yang YB (2016) Image restoration using convolutional auto-encoders with symmetric skip connections

Nah S, Kim TH, Lee KM (2017) Deep multi-scale convolutional neural network for dynamic scene deblurring, pp 257–265

Ni Z, Ma L, Zeng H, Cai C, Ma KK (2016) Gradient direction for screen content image quality assessment. IEEE Signal Process Lett 23(10):1394–1398

Ni Z, Ma L, Zeng H, Jing C, Cai C, Ma KK (2017) ESIM: edge similarity for screen content image quality assessment. IEEE Trans Image Process A Publ IEEE Signal Process Soc 26(10):4818–4831

Ono S, Yamada I (2016) Color-line regularization for color artifact removal. IEEE Trans Comput Imaging 2(3):204–217

Osher S, Burger M, Goldfarb D, Yin W (2011) An iterative regularization method for total variation based image restoration. In: IEEE international conference on imaging systems and techniques, pp 170–175

Rudin LI, Osher S, Fatemi E (1992) Nonlinear total variation based noise removal algorithms. In: Eleventh international conference of the center for nonlinear studies on experimental mathematics: computational issues in nonlinear science: computational issues in nonlinear science, pp 259–268

Wang Y, Gong J, Zhang D, Gao C, Tian J, Zeng H (2011) Large disparity motion layer extraction via topological clustering. IEEE Trans Image Process 20(1):43–52

Xie J, Xu L, Chen E (2012) Image denoising and inpainting with deep neural networks. In: International conference on neural information processing systems, pp 341–349

Yang A, Zeng H, Chen J, Zhu J, Cai C (2016) Perceptual feature guided rate distortion optimization for high efficiency video coding. Multidim Syst Signal Process 28(4):1–18

Zeng H, Ma KK, Cai C (2010) Hierarchical intra mode decision for H.264/AVC. IEEE Trans Circ Syst Video Technol 20(6):907–912

Zeng H, Ma KK, Cai C (2011) Fast mode decision for multiview video coding using mode correlation. IEEE Press, Oxford

Zeng H, Wang X, Cai C, Chen J (2014) Fast multiview video coding using adaptive prediction structure and hierarchical mode decision. IEEE Trans Circ Syst Video Technol 24(9):1566–1578

Zha Z, Liu X, Huang X, Shi H, Xu Y, Wang Q, Tang L, Zhang X (2017) Analyzing the group sparsity based on the rank minimization methods. In: IEEE international conference on multimedia and expo, pp 883–888

Zhang H (2016) Convolutional sparse coding-based image decomposition. In: British machine vision conference

Zhang K, Zuo W, Chen Y, Meng D, Zhang L (2017) Beyond a gaussian denoiser: residual learning of deep CNN for image denoising. IEEE Trans Image Process 26(7):3142–3155

Zhu J, Zeng H, Liao S, Lei Z, Cai C, Zheng LX (2017) Deep hybrid similarity learning for person re-identification. IEEE Trans Circ Syst Video Technol 99:1–1

Terms and Conditions

Springer Nature journal content, brought to you courtesy of Springer Nature Customer Service Center GmbH (“Springer Nature”).
Springer Nature supports a reasonable amount of sharing of research papers by authors, subscribers and authorised users (“Users”), for small-scale personal, non-commercial use provided that all copyright, trade and service marks and other proprietary notices are maintained. By accessing, sharing, receiving or otherwise using the Springer Nature journal content you agree to these terms of use (“Terms”). For these purposes, Springer Nature considers academic use (by researchers and students) to be non-commercial.
These Terms are supplementary and will apply in addition to any applicable website terms and conditions, a relevant site licence or a personal subscription. These Terms will prevail over any conflict or ambiguity with regards to the relevant terms, a site licence or a personal subscription (to the extent of the conflict or ambiguity only). For Creative Commons-licensed articles, the terms of the Creative Commons license used will apply.
We collect and use personal data to provide access to the Springer Nature journal content. We may also use these personal data internally within ResearchGate and Springer Nature and as agreed share it, in an anonymised way, for purposes of tracking, analysis and reporting. We will not otherwise disclose your personal data outside the ResearchGate or the Springer Nature group of companies unless we have your permission as detailed in the Privacy Policy.
While Users may use the Springer Nature journal content for small scale, personal non-commercial use, it is important to note that Users may not:

1. use such content for the purpose of providing other users with access on a regular or large scale basis or as a means to circumvent access control;
2. use such content where to do so would be considered a criminal or statutory offence in any jurisdiction, or gives rise to civil liability, or is otherwise unlawful;
3. falsely or misleadingly imply or suggest endorsement, approval , sponsorship, or association unless explicitly agreed to by Springer Nature in writing;
4. use bots or other automated methods to access the content or redirect messages
5. override any security feature or exclusionary protocol; or
6. share the content in order to create substitute for Springer Nature products or services or a systematic database of Springer Nature journal content.

In line with the restriction against commercial use, Springer Nature does not permit the creation of a product or service that creates revenue, royalties, rent or income from our content or its inclusion as part of a paid for service or for other commercial gain. Springer Nature journal content cannot be used for inter-library loans and librarians may not upload Springer Nature journal content on a large scale into their, or any other, institutional repository.
These terms of use are reviewed regularly and may be amended at any time. Springer Nature is not obligated to publish any information or content on this website and may remove it or features or functionality at our sole discretion, at any time with or without notice. Springer Nature may revoke this licence to you at any time and remove access to any copies of the Springer Nature journal content which have been saved.
To the fullest extent permitted by law, Springer Nature makes no warranties, representations or guarantees to Users, either express or implied with respect to the Springer nature journal content and all parties disclaim and waive any implied warranties or warranties imposed by law, including merchantability or fitness for any particular purpose.
Please note that these rights do not automatically extend to content, data or other material published by Springer Nature that may be licensed from third parties.
If you would like to use or distribute our Springer Nature journal content to a wider audience or on a regular basis or in any other manner not expressly permitted by these Terms, please contact Springer Nature at

onlineservice@springernature.com