Computers, Materials & Continua
DOI:10.32604/cmc.2022.020651
Article

Tech Science Press

Artifacts Reduction Using Multi-Scale Feature Attention Network in Compressed Medical Images

Seonjae Kim and Dongsan Jun[*]

Department of Convergence IT Engineering, Kyungnam University, Changwon, 51767, Korea
[*]Corresponding Author: Dongsan Jun. Email: dsjun9643@kyungnam.ac.kr
Received: 01 June 2021; Accepted: 03 July 2021

Abstract: Medical image compression is one of the essential technologies to facilitate real-time medical data transmission in remote healthcare applications. In general, image compression can introduce undesired coding artifacts, such as blocking artifacts and ringing effects. In this paper, we proposed a Multi-Scale Feature Attention Network (MSFAN) with two essential parts, which are multi-scale feature extraction layers and feature attention layers to efficiently remove coding artifacts of compressed medical images. Multi-scale feature extraction layers have four Feature Extraction (FE) blocks. Each FE block consists of five convolution layers and one CA block for weighted skip connection. In order to optimize the proposed network architectures, a variety of verification tests were conducted using validation dataset. We used Computer Vision Center-Clinic Database (CVC-ClinicDB) consisting of 612 colonoscopy medical images to evaluate the enhancement of image restoration. The proposed MSFAN can achieve improved PSNR gains as high as 0.25 and 0.24 dB on average compared to DnCNN and DCSC, respectively.

Keywords: Medical image processing; convolutional neural network; deep learning; telemedicine; artifact reduction; image restoration

1 Introduction

In the telemedicine field, a large number of medical images are produced from endoscopy, Computed Tomography (CT), and Magnetic Resonance Imaging (MRI). As these medical images have to support high quality to identify more accurate medical diagnoses, image compression is one of the essential technologies to facilitate real-time medical data transmission in remote healthcare applications. Although the latest image compression method can provide powerful coding performance without noticeable quality loss, both diagnostic uncertainty and degradation of subjective quality can be caused by image compression from a low bitrate environment with limited network bandwidth. In general, image compression can introduce undesired coding artifacts such as blocking artifacts and ringing effects primarily due to block-based coding to remove high-frequency components [1]. Because these artifacts can decrease perceptual visual quality, there is a need to reduce them on compressed medical images.

Deep learning methods using Convolutional Neural Network (CNN) have brought great potentials into low-level computer vision applications such as Super Resolution (SR) [2–8], image denoising [9–16], and image colorization [17,18]. In particular, these applications have been developed by CNN-based image denoising methods with deeper and denser network architectures [19,20]. Recently, these methods tend to be more complicated network architectures with enormous network parameters, excessive convolution operations, and high memory usages. In addition, most networks were initially designed to remove coding artifacts of natural images, so direct applications of them to medical images will lead to unsatisfactory performance. In this paper, we proposed a novel CNN structure to efficiently improve the quality of compressed medical images as shown in Fig. 1. The main contributions of this paper are summarized as follows:

- In order to reduce coding artifact of compressed medical image, we proposed a Multi-Scale Feature Attention Network (MSFAN) with two essential parts, which are multi-scale feature extraction layers and feature attention layers.
- Through a variety of ablation works, the proposed network architecture was verified to guarantee its optimal performance for coding artifact reduction.
- Finally, we evaluated the performance of image restoration on natural images as well as medical images to demonstrate versatile applications of the proposed MSFAN.

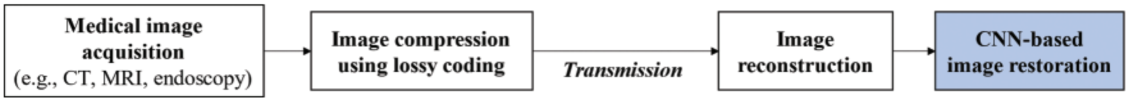
Medical image acquisition (e.g., CT, MRI, endoscopy)
Image compression using lossy coding
Transmission
Image reconstruction
CNN-based image restoration

Figure 1: CNN-based image restoration for coding artifact reduction in compressed medical images

The remainder of this paper is organized as follows. In Section 2, we review the previous CNN-based image restoration methods to remove the coding artifacts. Then, the proposed method is then described in Section 3. Finally, experimental results and conclusions are given in Sections 4 and 5, respectively.

2 Related Works

With the advancement of deep learning algorithms, the researches of low-level computer vision such as SR and image denoising has been combined with various CNN architectures to achieve higher image restoration. In the area of SR, Dong et al. have proposed a Super Resolution Convolutional Neural Network (SRCNN) [2] consisting of three convolutional layers. SRCNN can learn end-to-end pixel mapping from an interpolated low-resolution image to a high-resolution image. Since the advent of SRCNN, CNN-based image restoration methods have been reported with various deep learning models [21–27].

In terms of artifact reduction of compressed images, those methods can be applied to compressed images to reduce coding artifacts. As SR networks have generally up-sampling layers, the size of the output image is larger than that of the input image. On the other hand, the size of the output image is the same as that of the input image in the image denoising networks. Dong *et al.* have also proposed an Artifacts Reduction CNN (ARCNN) to reduce the coding artifacts compressed by Joint Photographic Experts Group (JPEG) [9]. Chen *et al.* addressed a

Trainable Nonlinear Reaction Diffusion (TNRD) for a variety of image restoration tasks, such as Gaussian image denoising, SR, and JPEG deblocking [10]. Zhang *et al.* have proposed a Denoising CNN (DnCNN) utilizing residual learning [21] and batch normalization [27] to enhance network training as well as denoising performance [11]. Fu *et al.* have proposed a Deep Convolutional Sparse Coding (DCSC) [13] to exploit multi-scale image features using three different dilated convolutions [25].

In terms of artifact reduction of compressed video sequences, CNN based video restoration methods show better performance than the conventional method. Lee *et al.* have proposed an algorithm to remove color artifacts using block-level quantization parameter offset control in compressed High Dynamic Range (HDR) videos [28]. On the other hand, Dai et al. have proposed CNN based video restoration, namely Variable-filter-size Residue-learning CNN (VRCNN) [14], which can be applied to compressed images by High Efficiency Video Coding (HEVC) [29]. Compared to ARCNN, this method can improve PSNR and reduce the number of parameters using small filter size. Meng *et al.* have proposed a Multi-channel Long-Short-term Dependency Residual Network (MLSDRN), which updates each cell to adaptively store and select long-term and short-term dependency information in HEVC [15]. Aforementioned image and video denoising networks can be deployed in the preprocessing of various high-level computer vision applications, such as object recognition [30–32] and detection [33,34] to achieve higher accuracy.

As depicted in Fig. 2, Hu *et al.* have presented a Channel Attention (CA) block, namely Squeeze-and-Excitation Network (SENet), which adaptively recalibrates channel-wise feature responses to represent interdependencies between feature maps [26], where GAP, F_i, and F_i^{CA} indicate global average polling operation, input feature maps of CA block and output feature maps, respectively. Zhang *et al.* have proposed a very deep Residual Channel Attention Network (RCAN), which deploys a CA block to adaptively rescale channel-wise features for improving SR performance [8]. Ding *et al.* have proposed a Squeeze-and-Excitation Filtering CNN (SEFCNN) to fully explore the relationship between channels in HEVC in-loop filter [16].

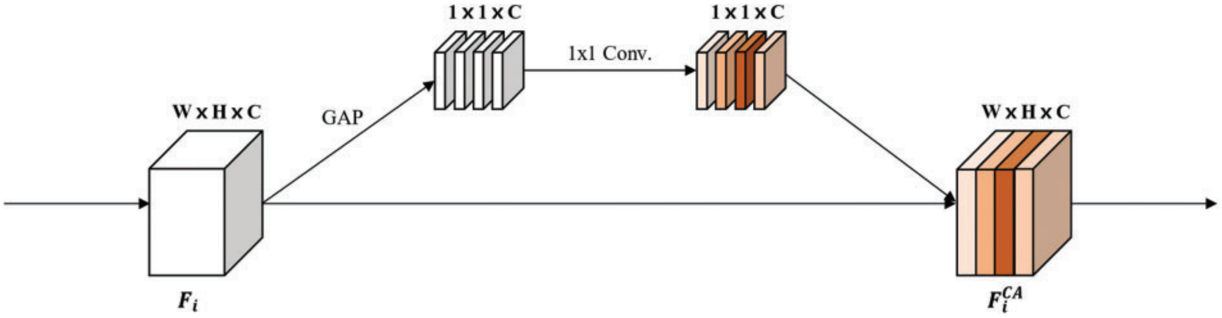

Figure 2: Architecture of the CA block [26] to assign different weights for each feature map

3 Proposed Methods

3.1 Overall Architecture of MSFAN

Fig. 3 shows the overall architecture of the proposed Multi-Scale Feature Attention Network (MSFAN) to remove coding artifacts in compressed medical images. It consists of an input layer, multi-scale feature extraction layers, feature attention layers, and an output layer. The convolutional operation of MSFAN calculates output feature maps (F_i) from previous feature maps (F_{i-1}) as expressed in Eq. (1):

$$F_i = \delta_i(W_i * F_{i-1} + B_i) \tag{1}$$

where $\delta_i(\cdot)$, W_i, B_i, and '$*$' represent Parametric Rectified Linear Unit (PReLU) function as an activation function, filter weights, biases, and convolutional operation, respectively. For fast and stable network training, the proposed MSFAN uses a residual learning scheme with skip connections [21]. Specifically, the input image is added to the feature map of the output layer using a skip connection to learn the residual image. In addition, output feature maps of the input layer are added to output feature maps of feature attention layers using CA-based weighted skip connections [26].

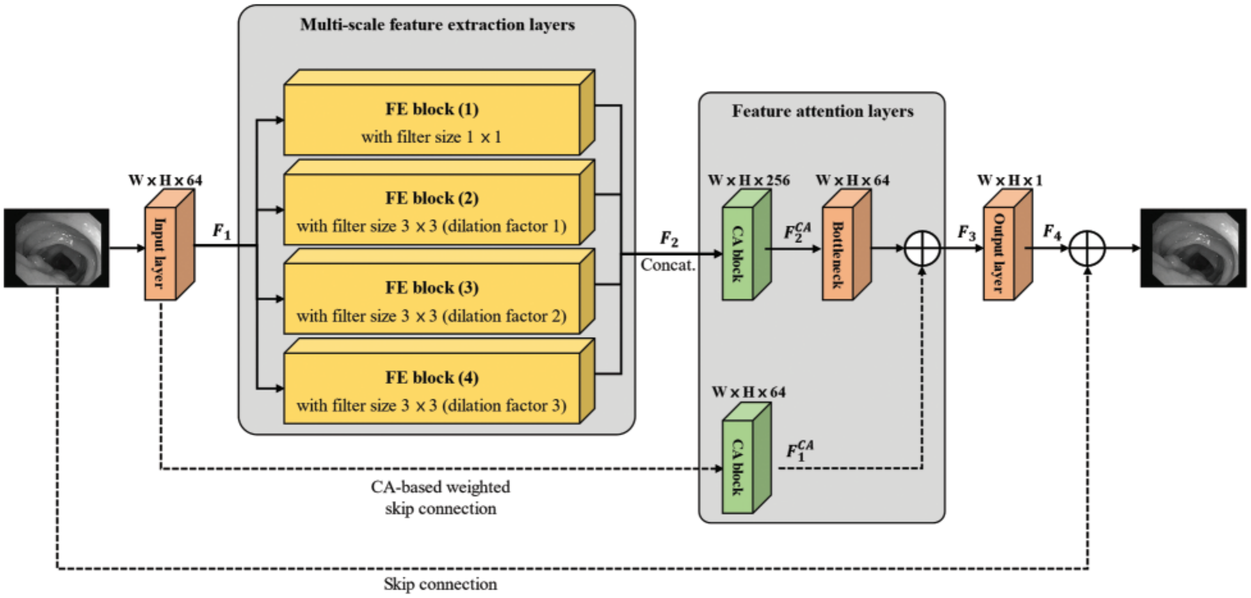

Figure 3: Overall architecture of the proposed MSFAN where symbol '⊕' indicates element-wise sum

As shown in Fig. 4a, the CA block consists of GAP and two convolutional layers. Because the CA block can emphasize more important feature maps for better network training, it assigns weights (H_i^{CA}) to each channel of input feature maps to adaptively control channel-wise feature response as expressed in Eq. (2):

$$H_i^{CA} = \sigma(W_2 * (W_1 * GAP(F_i) + B_1) + B_2) \tag{2}$$

where $\sigma(\cdot)$ indicates sigmoid function. Then, output feature maps (F_i^{CA}) of the CA block are generated from channel-wise product operations '$\otimes$' as shown in Eq. (3).

$$F_i^{CA} = F_i \otimes H_i^{CA} \tag{3}$$

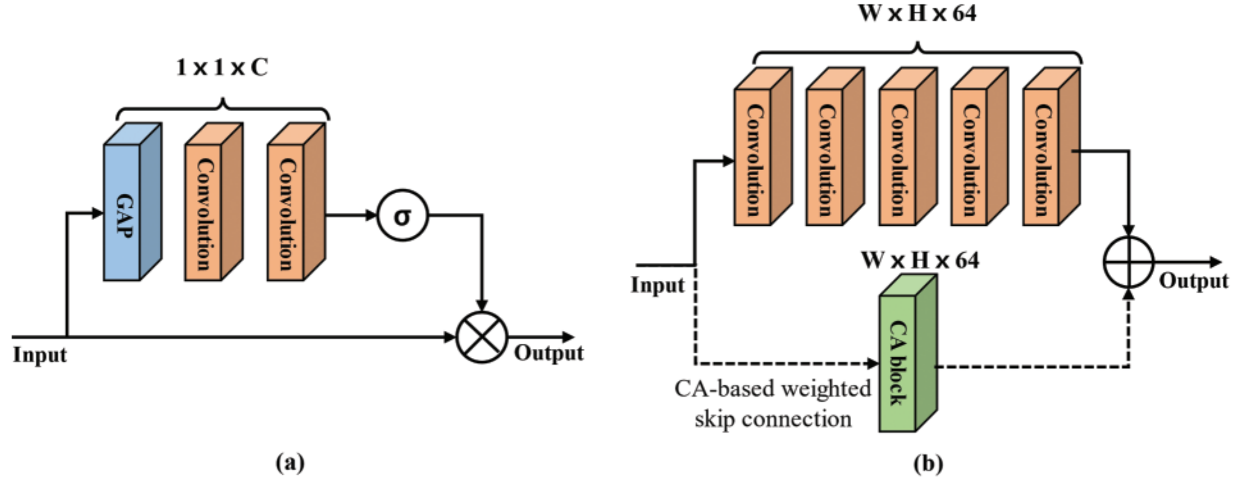

Figure 4: Structures of the (a) CA block and (b) FE block where symbols '**σ**' and '⊗' indicate sigmoid function and channel-wise product, respectively

Multi-scale feature extraction layers have four Feature Extraction (FE) blocks. Each FE block consists of five convolution layers and one CA block for weighted skip connection as shown in Fig. 4b. In the FE blocks, we used dilated convolutional operations with three different dilation facors (DF) to extract multi-scale features, as depicted in Fig. 5. Because large-size of filters will cause substantial increases for the number of parameters, we deployed dilated convolution to allow a wide receptive field without additional network parameters [25]. Note that CA-based weighted skip connection was also implemented on each FE block to train interdependencies between multi-scale channels.

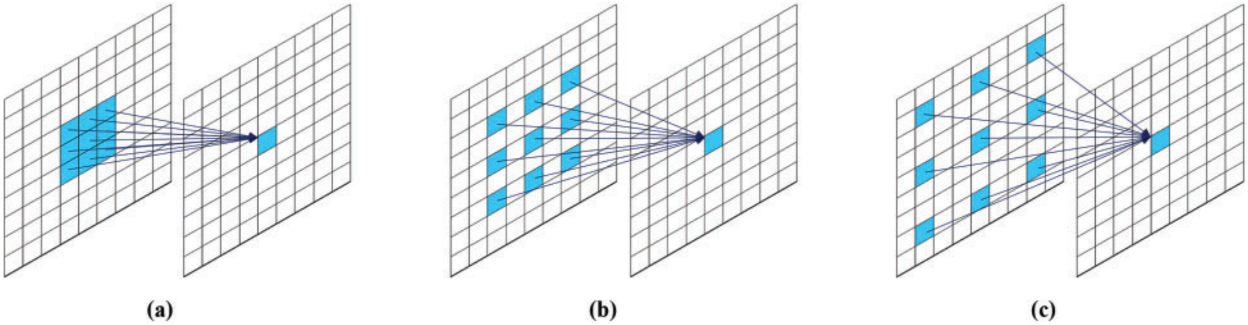

Figure 5: 3 × 3 dilated filters with different dilation factors to allow a wide receptive field without additional network parameters [25]. (a) 3 × 3 filter with dilation factor 1 (b) 3 × 3 filter with dilation factor 2 (c) 3 × 3 filter with dilation factor 3

In the feature attention layers, concatenated feature maps from all FE blocks (F_2) are used as the input of the next CA block. After generating output feature maps (F_2^{CA}) by CA block, they are fed into a bottleneck layer in order to reduce the number of output feature maps. It means that the bottleneck layer has a role in decreasing the number of filter weights as well as compressing the number of feature maps. Output feature maps (F_3) of the feature attention layers are computed by element-wise sum between F_1^{CA} and the output of the bottleneck layer. Finally, the output layer generates a predicted residual image (F_4) between the input and original images.

Note that we used zero padding to allow all feature maps to have the same spatial resolution between different convolutional layers, and the padding size is determined by Eq. (4):

$$Padding\ Size = \lfloor (Filter_W \times DF - (DF - 1))/2 \rfloor \tag{4}$$

where $Filter_W$ and $\lfloor \cdot \rfloor$ indicate the width of the filter and rounding down operation, respectively.

3.2 MSFAN Training

In order to find optimal network parameters, various hyper parameters are set as presented in Tab. 1. We used the loss function as expressed in Eq. (5) which represents regularization as well as Mean Square Error (MSE) as a data loss.

$$L(\theta) = \frac{1}{N} \sum_{i=0}^{N-1} ||O_i - Y_i||_2^2 + \frac{1}{2}\lambda ||\theta||_2^2 \tag{5}$$

Table 1: Hyper parameters used in the proposed MSFAN

Hyper parameter	Options
Loss function	Mean Square Error (MSE)
Optimizer	Adam [35]
Number of epochs	100
Batch size	128
Learning rate	10^{-4} to 10^{-6}
Weight decay factor	10^{-6}
Activation function	PReLU
Padding mode	Zero padding
Initial weight	Orthogonal [36]

In Eq. (5), θ, N, O_i, Y_i, and λ denote the set of network parameters (filter weights and biases), batch size, original image, restored image, and weight decay factor, respectively. Note that the proposed MSFAN used a weight decay scheme for network training to ensure generalization performance on various test datasets. In the training stage, the set of network parameters θ is updated using Adam optimizer [35] with a batch size of 128. In addition, filter weights are initialized by orthogonal normalization [36].

4 Experimental Results

All experiments were performed on an Intel Xeon Gold 5120 (14 cores @ 2.20 GHz) with 177 GB RAM and two NVIDIA Tesla V100 GPUs under the experimental environment described in Tab. 2. For performance comparison, the proposed MSFAN was compared with ARCNN [9], DnCNN [11], and DCSC [13] in terms of image restoration and network complexity.

4.1 Performance Comparisons for Medical Images

In order to evaluate the enhancement of image restoration, we used Computer Vision Center-Clinic Database (CVC-ClinicDB) [37] consisting of 612 colonoscopy medical images. We randomly divided CVC-ClinicDB into a training dataset (315 images), a validation dataset (103 images), and a test dataset (194 images). Note that all images were converted from the YUV color format into only Y component and compressed by JPEG codec under four different quality factors (10, 20, 30, and 40) to produce various coding artifacts. As a pre-processing of the training dataset, we cropped edges of each training image to remove unnecessary boundaries and extracted training images with a size of 32 × 32 without overlap. As a result, we collected 96,768 patches from the training dataset.

Table 2: Experimental environment of the proposed MSFAN

Environment	Options
Input image size	32 × 32 × 1
Output image size	32 × 32 × 1
Operating system version	Ubuntu 16.04
CUDA version	10.1
Deep learning framework	Pytorch 1.4.0

To evaluate the enhancement of image restoration, we measured Peak Signal-to-Noise Ratio (PSNR) and Structural Similarity Index Measure (SSIM) [38] between original and restored images. As measured in Tabs. 3 and 4, the proposed MSFAN can achieve the improved PSNR gains as high as 0.25 and 0.24 dB on average compared to DnCNN and DCSC, respectively. In addition, the proposed MSFAN showed better SSIM result on average than the other methods. Fig. 6 shows examples of visual comparisons between the proposed MSFAN and previous methods using test datasets. For each image in Fig. 6, images of the second row represent the zoom-in for the area indicated by the red box. These results verified that the proposed network could recover structural information effectively and find more accurate textures than other methods.

Table 3: Average PSNR (dB) on CVC-ClinicDB test dataset where the best results of PSNR are shown in bold

Quality factor	JPEG [1]	ARCNN [9]	DnCNN [11]	DCSC [13]	MSFAN
10	33.42	33.98	35.36	35.13	**35.54**
20	36.34	36.89	37.79	**38.09**	38.04
30	37.81	38.47	39.32	39.21	**39.52**
40	39.08	39.34	39.98	40.06	**40.36**
Average	36.66	37.17	38.11	38.12	**38.36**

Table 4: Average SSIM on CVC-ClinicDB test dataset where the best results of SSIM are shown in bold

Quality factor	JPEG [1]	ARCNN [9]	DnCNN [11]	DCSC [13]	MSFAN
10	0.857	0.882	0.903	0.904	**0.906**
20	0.910	0.921	0.932	**0.934**	0.933
30	0.932	0.942	0.946	0.947	**0.948**
40	0.945	0.953	0.954	**0.956**	**0.956**
Average	0.911	0.925	0.934	0.935	**0.936**

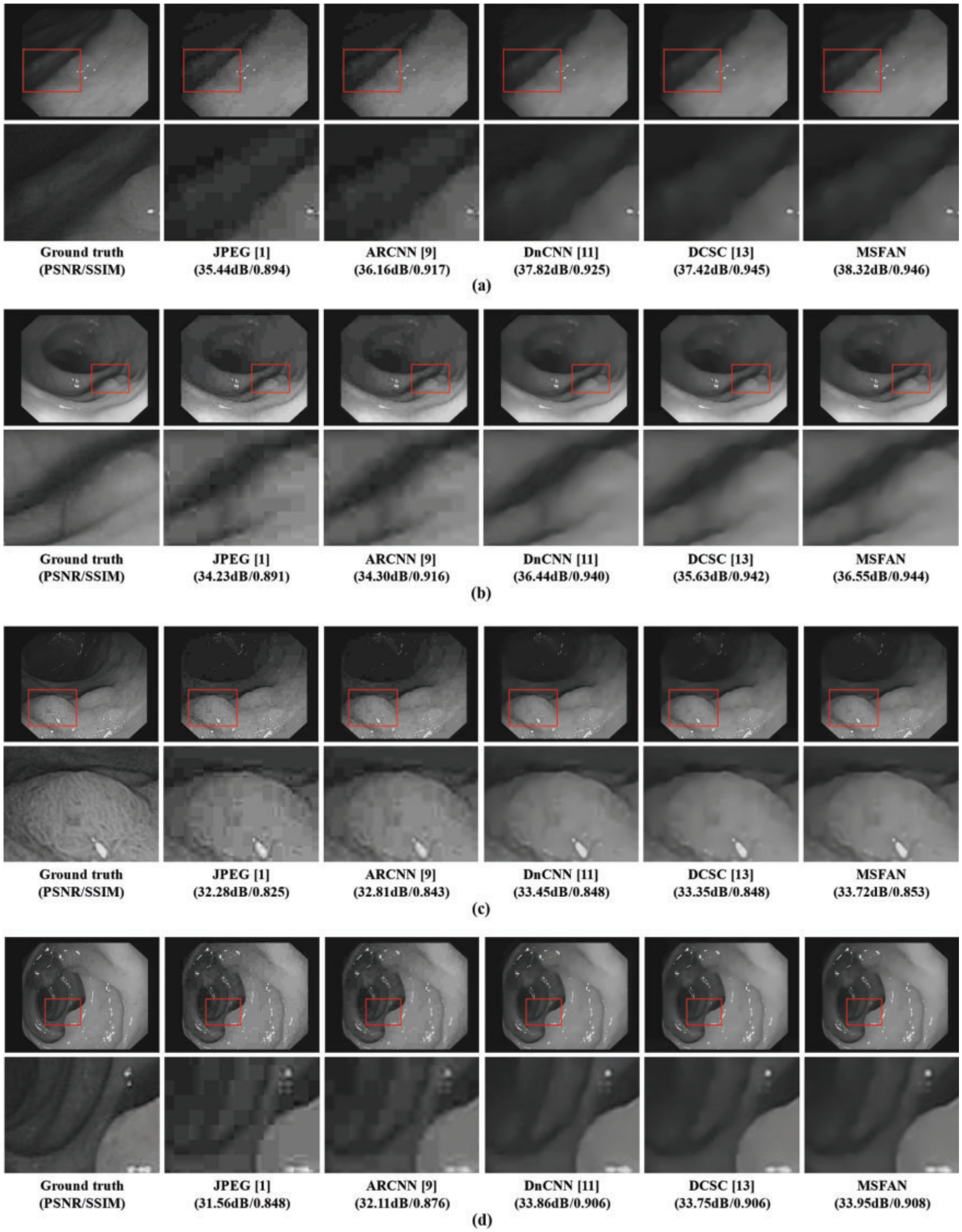
Ground truth (PSNR/SSIM)
JPEG [1] (35.44dB/0.894)
ARCNN [9] (36.16dB/0.917)
DnCNN [11] (37.82dB/0.925)
DCSC [13] (37.42dB/0.945)
MSFAN (38.32dB/0.946)
(a)
Ground truth (PSNR/SSIM)
JPEG [1] (34.23dB/0.891)
ARCNN [9] (34.30dB/0.916)
DnCNN [11] (36.44dB/0.940)
DCSC [13] (35.63dB/0.942)
MSFAN (36.55dB/0.944)
(b)
Ground truth (PSNR/SSIM)
JPEG [1] (32.28dB/0.825)
ARCNN [9] (32.81dB/0.843)
DnCNN [11] (33.45dB/0.848)
DCSC [13] (33.35dB/0.848)
MSFAN (33.72dB/0.853)
(c)
Ground truth (PSNR/SSIM)
JPEG [1] (31.56dB/0.848)
ARCNN [9] (32.11dB/0.876)
DnCNN [11] (33.86dB/0.906)
DCSC [13] (33.75dB/0.906)
MSFAN (33.95dB/0.908)
(d)

Figure 6: Visual comparisons of medical images where figures of the second row represent the zoom in for the area indicated by the red box of the first row

4.2 Performance Comparisons for Natural Images

We further evaluated the proposed MSFAN for natural images to demonstrate versatile applications of our network. For training the MSFAN with the natural image dataset, we used 400 images from BSD500 [39]. Similar to medical images, all training images were converted into YUV color format and only Y components were extracted with a size of 32 × 32 using data agumentation including rotation and flip. For the test image dataset, we used Classic5 which commonly used as testing dataset in various image restoration studies [19,20]. Tabs. 5 and 6 show average PSNR and SSIM results on Classic 5, respectively. While the proposed MSFAN had marginally lower PSNR values than DnCNN on average, these SSIM results were superior to those of comparison networks except that JPEG quality factor was 10.

Table 5: Average PSNR (dB) on Classic5 dataset where the best results of PSNR are shown in bold

Quality factor	JPEG [1]	ARCNN [9]	DnCNN [11]	DCSC [13]	MSFAN
10	27.82	29.03	**29.40**	29.25	29.39
20	30.12	31.15	**31.63**	31.43	31.55
30	31.48	32.51	**32.91**	32.68	32.85
Average	29.81	30.90	**31.31**	31.12	31.26

Table 6: Average SSIM on Classic5 dataset where the best results of SSIM are shown in bold

Quality factor	JPEG [1]	ARCNN [9]	DnCNN [11]	DCSC [13]	MSFAN
10	0.769	0.792	**0.886**	0.803	0.811
20	0.845	0.852	0.861	0.860	**0.868**
30	0.876	0.881	0.886	0.885	**0.893**
Average	0.830	0.842	**0.878**	0.849	0.857

4.3 Ablation Studies

In order to optimize the proposed network architecture, we conducted a variety of verification tests using the validation dataset. First, we performed tool-off tests to verify the effectiveness of essential parts of the proposed network, as shown in Tab. 7. According to the results of tool-off tests confirmed that both FE and CA blocks have an effect on the performance of image restoration. Additionally, we investigated two verification tests to determine optimal number of channels and 1 × 1 convolutional layers in the CA block. Tabs. 8 and 9 show that the proposed MSFAN has an optimal network architecture.

4.4 Computational Complexity

In order to investigate network complexity, we analyzed the number of parameters, total memory size, and inference speed using the test dataset. Note that the total memory size denotes the amount of memory required to store both network parameters and feature maps. As shown in Tab. 10, the proposed MSFAN has smaller total memory size than both DnCNN and DCSC, while it has more network parameters than other methods. In addition, Fig. 7 shows that the inference speed of our network is almost similar to that of DCSC using the CVC-ClinicDB test dataset.

Table 7: Verification tests for the effectiveness of FE blocks, CA block, and skip connection

Category	PSNR (dB)
MSFAN	35.81
FE block (1) off	35.57
FE block (2) off	35.63
FE block (3) off	35.64
FE block (4) off	35.71
CA block off	35.72
Skip connection off	35.10

Table 8: Verification tests for the number of channels of the output feature map

Category	PSNR (dB)
64 channels (MSFAN)	35.81
32 channels	35.60
128 channels	35.60
192 channels	35.70

Table 9: Verification tests on the number of 1 × 1 convolutional layers in the CA block

Category	PSNR (dB)
2 layers (MSFAN)	35.81
3 layers	35.63
4 layers	35.60

Table 10: Comparisons of network complexity between the proposed MSFAN and previous methods

	ARCNN [9]	DnCNN [11]	DCSC [13]	MSFAN
Number of parameters	106,561	667,072	93,697	765,587
Total memory size (MB)	2.17	30.56	67.60	29.45

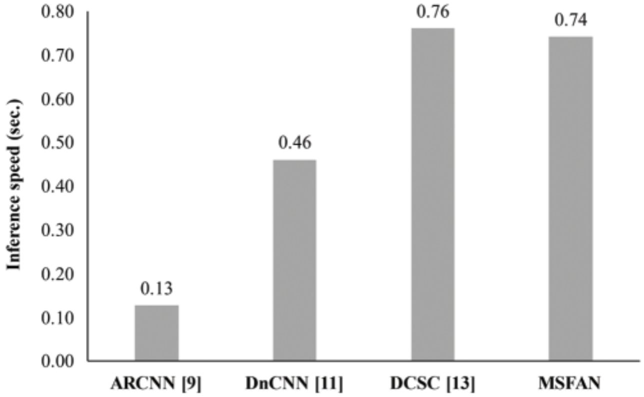

Figure 7: Comparisons of inference speed between the proposed MSFAN and previous methods

5 Conclusions

Medical image compression is one of the essential technologies to facilitate real-time medical data transmission in the remote healthcare applications. In general, image compression is known to introduce undesired coding artifacts, such as blocking artifacts and ringing effects. In this paper, we proposed a Multi-Scale Feature Attention Network (MSFAN) with two essential parts, which are multi-scale feature extraction layers and feature attention layers to efficiently remove the coding artifacts of compressed medical images. Multi-scale feature extraction layers have four Feature Extraction (FE) blocks, and each FE block consists of five convolution layers and one CA block for weighted skip connection. In order to optimize the proposed network architecture, we conducted a variety of verification tests using the validation dataset. We used Computer Vision Center-Clinic Database (CVC-ClinicDB) consisting of 612 colonoscopy medical images to evaluate the enhancement of image restoration. The proposed MSFAN can improve PSNR gains as high as 0.25 and 0.24 dB on average compared to DnCNN and DCSC, respectively.

Acknowledgement: This work was supported by Kyungnam University Foundation Grant, 2020.

Funding Statement: This work was supported by Kyungnam University Foundation Grant, 2020.

Conflicts of Interest: The author declare that they have no conflicts of interest to report regarding the present study.

References

[1] G. K. Wallace, "The JPEG still picture compression standard," *IEEE Transactions on Consumer Electronics*, vol. 38, no. 1, pp. 18–34, 1992.

[2] C. Dong, C. C. Loy, K. He and X. Tang, "Image super-resolution using deep convolutional networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 38, no. 2, pp. 295–307, 2015.

[3] J. Kim, J. K. Lee and K. M. Lee, "Accurate image super-resolution using very deep convolutional networks," in *Proc. CVPR*, Las Vegas, NV, USA, pp. 1646–1654, 2016.

[4] B. Lim, S. Son, H. Kim, S. Nah and K. M. Lee, "Enhanced deep residual networks for single image super-resolution," in *Proc. CVPRW*, Honolulu, HI, USA, pp. 136–144, 2017.

[5] T. Tong, G. Li, X. Liu and Q. Gao, "Image super-resolution using dense skip connections," in *Proc. ICCV*, Venice, Italy, pp. 4799–4807, 2017.

[6] C. Ledig, L. Theis, F. Huszár, C. Ferenc, J. Caballero *et al.*, "Photo-realistic single image super-resolution using a generative adversarial network," in *Proc. CVPR*, Honolulu, HI, USA, pp. 4681–4690, 2017.

[7] Y. Zhang, Y. Tian, Y. Kong, B. Zhong and Y. Fu, "Residual dense network for image super-resolution," in *Proc. CVPR*, Salt Lake City, UT, USA, pp. 2472–2481, 2018.

[8] Y. Zhang, K. Li, L. Kai, B. Zhong and Y. Fu, "Image super-resolution using very deep residual channel attention networks," in *Proc. ECCV*, Munich, Germany, pp. 286–301, 2018.

[9] C. Dong, Y. Deng, C. C. Loy and X. Tang, "Compression artifacts reduction by a deep convolutional network," in *Proc. ICCV*, Santiago, Chile, pp. 576–584, 2015.

[10] Y. Chen and T. Pock, "Trainable nonlinear reaction diffusion: A flexible framework for fast and effective image restoration," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 6, pp. 1256–1272, 2016.

[11] K. Zhang, W. Zuo, Y. Chen, D. Meng and L. Zhang, "Beyond a Gaussian denoiser: Residual learning of deep cnn for image denoising," *IEEE Transactions on Image Processing*, vol. 26, no. 7, pp. 3142–3155, 2017.

[12] X. Zhang, W. Yang, Y. Hu and J. Liu, "DMCNN: Dual-domain multi-scale convolutional neural network for compression artifacts removal," in *Proc. ICIP*, Athens, Greece, pp. 390–394, 2018.

[13] X. Fu, Z. J. Zha, F. W, X. Ding and J. Paisley, "Jpeg artifacts reduction via deep convolutional sparse coding," in *Proc. ICCV*, Seoul, Korea, pp. 2501–2510, 2019.

[14] Y. Dai, D. Liu and F. Wu, "A convolutional neural network approach for post-processing in HEVC intra coding," in *Proc. MMM*, Pittsburgh, PA, USA, pp. 28–39, 2017.

[15] X. Meng, C. Chen, S. Zhu and B. Zeng, "A new HEVC in-loop filter based on multi-channel long-short-term dependency residual networks," in *Proc. DCC*, Snowbird, UT, USA, pp. 187–196, 2018.

[16] D. Ding, L. Kong, G. Chen, Z. Liu and Y. Fang, "A switchable deep learning approach for in-loop filtering in video coding," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 30, no. 7, pp. 1871–1887, 2019.

[17] S. Anwar, M. Tahir, C. Li, A. Mian, F. S. Khan *et al.*, "Image colorization: A survey and dataset," arXiv preprint, 2020. https://arxiv.org/abs/2008.10774.

[18] A. Popwicz and B. Smolka, "Overview of grayscale image colorization techniques," in *Color Image and Video Enhancement*, 1^{st} ed., vol. 1. Berlin, Germany: Springer, pp. 345–370, 2015.

[19] J. Liu, D. Liu, W. Yang, S. Xia, X. Zhang *et al.*, "A comprehensive benchmark for single image compression artifact reduction," *IEEE Transactions on Image Processing*, vol. 29, pp. 7845–7860, 2020.

[20] C. Tian, L. Fei, W. Zheng, Y. Xu, W. Zuo *et al.*, "Deep learning on image denoising: An overview," *Neural Networks*, vol. 131, pp. 251–275, 2020.

[21] K. He, X. Zhang, S. Ren and J. Sun, "Deep residual learning for image recognition," in *Proc. CVPR*, Las Vegas, NV, USA, pp. 770–778, 2016.

[22] S. Huang, Z. Liu, L. V. D. Maaten and K. Q. Weinberger, "Densely connected convolutional networks," in *Proc. CVPR*, Honolulu, HI, USA, pp. 4700–4708, 2017.

[23] J. Kim, J. Kim, H. L. T. Thu and H. Kim, "Long short term memory recurrent neural network classifier for intrusion detection," in *Proc. PlatCon*, Jeju, Korea, pp. 1–5, 2016.

[24] I. Sutskever, O. Vinyals and Q. V. Le, "Sequence to sequence learning with neural networks," arXiv preprint, 2014. https://arxiv.org/abs/1409.3215.

[25] F. Yu and V. Koltun, "Multi-scale context aggregation by dilated convolutions," arXiv preprint, 2015. https://arxiv.org/abs/1511.07122.

[26] J. Hu, L. Shen and G. Sun, "Squeeze-and-excitation networks," in *Proc. CVPR*, Salt Lake City, UT, USA, pp. 7132–7141, 2018.

[27] S. Ioffe and C. Szegedy, "Batch normalization: accelerating deep network training by reducing internal covariate shift," in *Proc. ICML*, Lille, France, pp. 448–456, 2015.

[28] J. H. Lee, Y. W. Lee, D. Jun and B. G. Kim, "Efficient color artifact removal algorithm based on high-efficiency video coding (HEVC) for high-dynamic range video sequences," *IEEE Access*, vol. 8, pp. 64099–64111, 2020.

[29] G. J. Sullivan, J. R. Ohm, W. J. Han and T. Wiegand, "Overview of the high efficiency video coding (HEVC) standard," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 22, no. 12, pp. 1649–1668, 2012.

[30] J. H. Kim, G. S. Hong, B. G. Kim and D. P. Dogra, "Deepgesture: Deep learning-based gesture recognition scheme using motion sensors," *Display*, vol. 55, pp. 38–45, 2018.

[31] J. H. Kim, B. G. Kim, P. P. Roy and D. M. Jeong, "Efficient facial expression recognition algorithm based on hierarchical deep neural network structure," *IEEE Access*, vol. 7, pp. 41273–41285, 2019.

[32] D. Jeong, B. G. Kim and S. Y. Dong, "Deep joint spatiotemporal network (DJSTN) for efficient facial expression recognition," *Sensors*, vol. 20, no. 7, pp. 1936, 2020.

[33] S. Mukherjee, R. Saini, P. Kumar, P. P. Roy, D. Dogra *et al.*, "Fight detection in hockey videos using deep network," *Journal of Multimedia Information System*, vol. 4, no. 4, pp. 225–232, 2017.

[34] M. Chhetri, S. Kumar, P. P. Roy and B. G. Kim, "Deep BLSTM-gRU model for monthly rainfall prediction: A case study of simtokha, Bhutan," *Remote Sensing*, vol. 12, no. 19, pp. 3174, 2020.

[35] D. P. Kingma and K. Ba, "Adam: A method for stochastic optimization," arXiv preprint, 2014. https://arxiv.org/abs/1412.6980.

[36] A. M. Saxe, J. L. McClelland, L. James and S. Ganguli, "Exact solutions to the nonlinear dynamics of learning in deep linear neural networks," arXiv preprint, 2013. https://arxiv.org/abs/1312.6120.

[37] J. Bernal, F. J. Sánchez, G. Fernández-Esparrach, D. Gil, C. Rodriguez *et al.*, "WM-Dova maps for accurate polyp highlighting in colonoscopy: Validation vs. saliency maps from physicians," *Computerized Medical Imaging and Graphics*, vol. 43, pp. 99–111, 2015.

[38] Z. Wang, A. C. Bovik, H. R. Sheikh and E. P. Simoncelli, "Image quality assessment: From error visibility to structural similarity," *IEEE Transactions on Image Processing*, vol. 13, no. 4, pp. 600–612, 2004.

[39] D. Martin, C. Fowlkes, D. Tal and J. Malik, "A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics," in *Proc. ICCV*, Vancouver, Canada, pp. 416–423, 2001.