ORIGINAL ARTICLE

Multi-scale network toward real-world image denoising

Lianmin Zhou[1] · Dongming Zhou[1] · Hao Yang[1] · Shaoliang Yang[1]

Abstract

Images are inevitably degraded when captured due to the effects of noise, and thus denoising is required. Previous methods remove real-world noise, while also causing issues with over-smoothing image details and loss of edge information. To solve these issues, a multi-scale image denoising network (MSIDNet) is proposed in this paper. We design a residual attention block (RAB) to encode and decode the context well, while introducing a selective kernel feature fusion module to fuse multi-scale features and obtain rich contextual information from low-resolutions to restore more details. A feature extraction block (FEB) is designed to fully extract local and global features then fusion, which obtains rich feature information. Extensive experiments on four real-world image datasets demonstrate that our method has excellent generalization and achieves advanced denoising performance on both peak signal-to-noise ratio and structural similarity. MSIDNet preserves more edge details and improves the over-smoothing issue to enhance the visual effect of denoised images.

Keywords Image denoising · Real-world · Multi-scale · Feature extraction · Residual learning

1 Introduction

Image denoising is a fundamental task in image processing. To acquire clean images, we need to remove noise from the degraded images [1]. The increased number of image capture devices and the use of different sensor sizes can introduce noise to varying degrees. For example, mobile phone apertures are small and narrow, which tend to generate noise. The effectiveness of denoising influences the image quality, also benefits other computer vision works [2, 3] and computational tasks [4–7]. Image denoising has undergone a long development and is mainly divided into traditional denoising methods and deep learning-based denoising methods. Traditional denoising methods in the pre-development period needed a priori information about the noise and were optimized by manually adjusting the parameters of the model [8–10]. Therefore, the traditional denoising methods had to consume a large number of computational resources and time, while the denoising performance was general. As the

research of deep learning in image denoising, the denoising performance of images has been improved [11–13]. However, in these works, the experiments were performed using synthetic noisy images, whereas real-world noise is usually superimposed by multiple types, and the noise characteristics and distribution are unknown. For real-world noisy images, some research works have achieved good results [14–21]. The recent state-of-the-art methods have further improved the denoising performance [22, 23]. These methods enhance the denoising capability, but cause problems of over-smoothing image details and loss of edge information.

In this paper, we propose a multi-scale image denoising network to solve the aforementioned problems. We design a residual attention block to improve context-awareness. The triplet attention [24] is further introduced in each residual attention block to adjust the feature weights so that the network focuses more on the informative features. To make full use of global and local features, we design a feature extraction block combining the transformer block [25] and the residual block [26], and introduce a selective kernel feature fusion module [27] to effectively fuse features. Compared with other denoising methods, our method achieves superior denoising performance. In conclusion, our main contributions in this work include:

✉ Dongming Zhou
  zhoudm@ynu.edu.cn

  Lianmin Zhou
  zhou_lianmin@mail.ynu.edu.cn

[1] School of Information Science and Engineering, Yunnan University, Kunming 650504, China

- We propose a multi-scale image denoising network to remove noise effectively. The network is able to restore details from complex real-world noisy images by utilizing rich feature information.
- We propose the residual attention block to encode and decode the context, and we also employ triplet attention to obtain a more refined flow. In addition, a feature extraction block is further designed to obtain global and local features that complement each other.
- We conduct experiments qualitatively and quantitatively on four real-world noisy datasets. Experimental results demonstrate that our model achieves excellent denoising performance and generalization on all four datasets.

The rest of this paper is structured as follows. Section 2 reviews common denoising methods. Section 3 presents our proposed denoising network in detail. In Sect. 4, we conduct quantitative and qualitative experiments on multiple datasets. Finally, we conclude this work.

2 Related work

2.1 Traditional denoising methods

Traditional denoising methods played an important role in the early stage. Representative methods such as NLM [8] utilized non-local self-similarity to iterate over all pixel points of an image for denoising. Combined with the above non-local methods, further research by Dabov et al. proposed BM3D [9], which integrated several similar patches into a three-dimensional matrix by matching adjacent image patches and then conducted denoising. TWSC [10] was coded by three weight matrices for noise removal. The data used in these methods were synthetic noisy images. Although these traditional denoising methods were enhanced, the manual adjustment of the parameters and the complexity of the optimization algorithms consumed a large amount of time and computational resources.

2.2 Deep learning-based denoising methods

In recent years, research on denoising algorithms based on deep learning techniques has developed rapidly. Some works [11–13] had tried to apply deep learning to the field of image denoising. These methods were still researched for specific types of synthetic noise, such as Gaussian noise. While these methods are able to fit most noise distributions, they do not effectively remove complex noise.

With further research in deep learning, several works [14–19] made progress in real-world noise. By capturing real-world noisy scene images to establish several noisy datasets [28–31], this facilitated the research of real-world

image denoising. Path-Restore [20] used multiple path strategies to dynamically restore different areas of the image. COLA-Net [21] worked with multiple attention mechanisms to restore the complex texture of an image. The recent state-of-the-art methods, VDIR [22] and LIGN [23], divide the feature regions and then process for different regions to restore images. However, these methods do not deal well with problems that the image edge information is lost and details are too smooth.

3 Proposed method

In this section, we first present the overall architecture of the proposed MSIDNet. Then, the residual attention block and feature extraction block in the proposed network are described in detail. Finally, we state the loss function.

3.1 Architecture of the proposed MSIDNet

To achieve denoising of real-world images, we propose a multi-scale image denoising network (MSIDNet), the architecture is shown in Fig. 1. The noisy image first passes to a convolutional layer and then enters the designed feature extraction block (FEB) to obtain global and local features, while introducing selective kernel feature fusion (SKFF) [27] to effectively fuse those features. SKFF is based on a self-attention mechanism to aggregate weighting features. Then, we design the residual attention block (RAB) for coding and decoding to remove the noise and reconstruct the image texture. After three down-sampling operations, the feature information at different scales is then fused progressively. Finally, the denoised image is obtained after a convolutional layer.

We implement down-sampling operations by using convolutions with kernel size two and step size two, and the number of channels is expanded with each down-sampling operation. The up-sampling operation is achieved by using PixelShuffle [32] and point-wise convolution. Using Transposed convolution for up-sampling operation will lead to checkerboard artifacts [33]. We solve this problem by using PixelShuffle to better restore the information.

3.2 Residual attention block

Figure 2 shows the structure of our proposed residual attention block. In RAB, the input feature map $f_i \in \mathbb{R}^{C \times H \times W}$ is first convolved by two 3×3 convolutions to obtain shallow features, and the GELU activation function is used for nonlinear projection, where W, H, C represent width, height, number of channels, respectively. The GELU activation function combines a stochastic regularity method that makes the network enhance generalization and solves the gradient
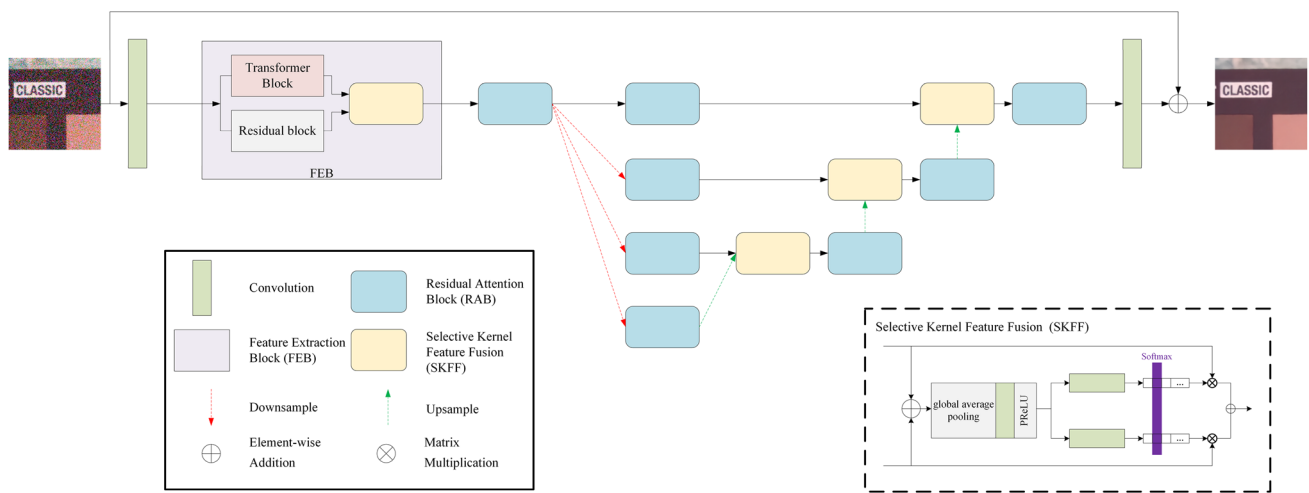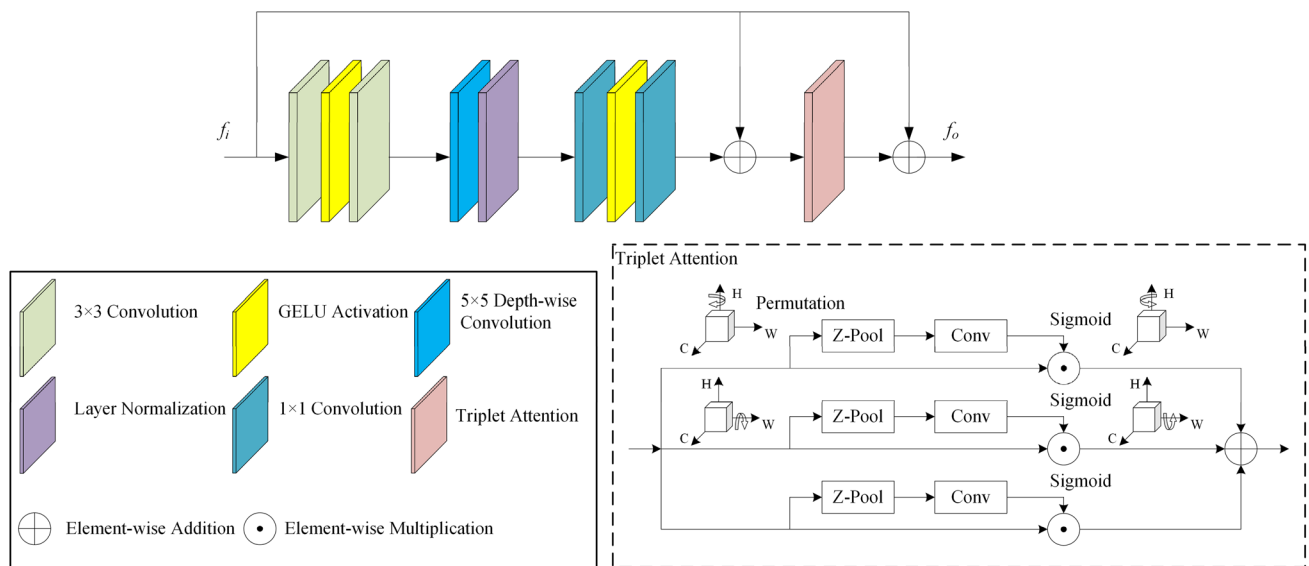
Fig. 1 Architecture of the proposed MSIDNet

Fig. 2 The detailed structure of RAB

disappearance problem [34]. Inspired by [35], we use a 5×5 depth-wise convolution to blend the information in f_i spatial dimensions while expanding the receptive field without excessively increasing the computational effort, and then conduct Layer Normalization. Layer Normalization alleviates the internal covariate shift problem and is not influenced by batch size [36]. Next, the information in each channel is blended by two point-wise convolutions and projected nonlinearly by the GELU activation function. Finally, we introduce triplet attention [24] to filter the more informative features to pass. Triplet attention consists of three branches, where the first two branches make the connection between the spatial dimension and the channel dimension by rotation

operations, and the third branch is responsible for calculating the spatial attention weights. We further exploit the structure of local residual learning as a way to bypass less useful information such as low-frequency regions. Finally, the result $f_o \in \mathbb{R}^{C\times H\times W}$ of the RAB processing is obtained. The whole process can be described as follows:

$$f_o = T(Conv1(LN(D(Conv3(f_i)))) \oplus f_i) \oplus f_i, \qquad (1)$$

where f_i and f_o denote input and output feature maps respectively, and $Conv3$ denotes two 3×3 convolutions, and D denotes the 5×5 depth-wise convolution, and LN denotes the Layer Normalization, and $Conv1$ denotes two point-wise

convolutions, and T denotes the triplet attention, and $\oplus$ denotes the element-wise addition.

3.3 Feature extraction block

The rich feature information facilitates real-world image noise removal. Transformer is excellent at global processing and has shown powerful advantages in tasks such as image classification [37] and image segmentation [38]. Convolutional neural networks employ convolutional operations for effective local processing.

To utilize local and global feature information effectively, we introduce the transformer block [25] and the residual block [26] to design a feature extraction block. The transformer block is used to obtain global features, the residual block for getting local features, and finally the feature information is fused by SKFF. The structure is shown in Fig. 1.

3.4 Loss function

We use Charbonnier loss [39] to optimize our MSIDNet. Charbonnier loss contains a regularization term β, which can serve to speed up network convergence and improve performance. The formula is as follows [39]:

$$L(I, I^*) = \sqrt{\|I - I^*\|^2 + \beta^2},$$ (2)

where I^* denotes the ground-truth image, and I denotes the denoised image. β is a constant representing the regularization term, which is empirically set to 10^{-3} for the experiments in this paper.

4 Experiments

In this section, the denoising performance of our proposed model is evaluated both quantitatively and qualitatively, and ablation study is used to verify the effectiveness of modules.

4.1 Evaluation metrics

We use two metrics, structural similarity (SSIM) and peak signal-to-noise ratio (PSNR) [40], to quantitatively analyze the model denoising performance. The higher the PSNR is, the closer the SSIM is to 1, indicating that the denoised image is more similar to the ground-truth image and the model performs better in denoising. The SSIM and PSNR formulas are as follows [40]:

$$PSNR = 20 \times \log_{10} \left(\frac{MAX[K(i,j)]}{\frac{1}{HW} \sum_{i=1}^{H} \sum_{j=1}^{W} [K(i,j) - P(i,j)]^2} \right),$$ (3)

$$SSIM = \frac{(2u_1 u_2 + n_1)(2\sigma_{1,2} + n_2)}{(u_1^2 + u_2^2 + n_1)(\sigma_1^2 + \sigma_2^2 + n_2)},$$ (4)

where W denotes the width, and H denotes the height, and $P(i, j)$ denotes the pixel values at the location of the denoised image (i, j), and $K(i, j)$ denotes the pixel values at the location of the ground-truth image (i, j), and u_1, u_2 denote the mean values of $K(i, j)$, $P(i, j)$, and σ_1, σ_2 denote the variance of $K(i, j)$, $P(i, j)$, and $\sigma_{1,2}$ denotes the covariance of $K(i, j)$, $P(i, j)$, and $n_1 = 0.01$ and $n_2 = 0.02$ are constants.

4.2 Experimental platform and datasets

Four real-world noisy image datasets SIDD [28], DND [29], PolyU [30], and RNI15 [31] are used to measure the network denoising performance. The SIDD dataset was captured by five different smartphones, which generate a lot of noise during image acquisition due to their small sensor size. The dataset has 320 noisy and noise-free image pairs for training the model, and 40 image pairs are cropped by the authors into 1280 pairs of 256×256 patches for testing the model's performance. The DND dataset was acquired by four different consumer-grade cameras and consists of fifty pairs of noisy and noise-free image pairs. This author crops large size images into 1000 patches of size 512×512 for testing and does not disclose the noise-free images, only by submitting denoising results to the official system to obtain SSIM and PSNR. The PolyU dataset was captured by five different cameras, and this author crops 40 pairs of noisy and noise-free image pairs into 100 pairs of 512×512 patches. RNI15 consists of 15 real-world noisy images with no corresponding noise-free images, so qualitative comparison experiments are performed on this dataset. We crop 96,000 pairs in size 256×256 patches from the SIDD training set for training our model.

The model denoising performance is tested using sRGB images from four real-world noisy datasets and the PSNR is calculated on the RGB channel. We use the Pytorch framework to build the network structure, and the main device used is the NVIDIA RTX 3080Ti. During training, we use the cosine annealing strategy to stabilize the learning rate reduction and optimize the network parameters with Charbonnier Loss and AdamW optimizer ($\beta_1 = 0.9$, and $\beta_2 = 0.999$), and our model is trained for a total of 80 epochs with the batch size of 12. The loss curve is shown in Fig. 3, where we can see that the loss value gradually decreases and stabilizes with increasing epoch number.

4.3 Quantitative comparison

We compare the denoising performance of our proposed model with thirteen excellent methods on four datasets,
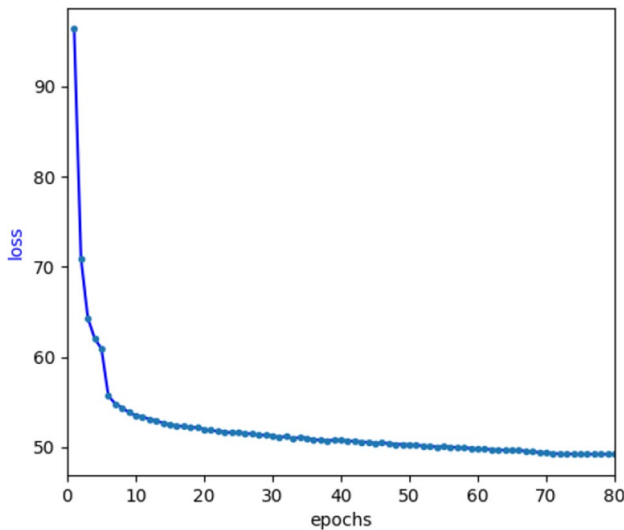
Fig. 3 Loss changes during training

among which ten blind denoising methods and three non-blind denoising methods are tested, including COLA-Net [21], HI-GAN [19], C2N [17], Path-Restore [20], CBDNet [15], FFDNet [14], TWSC [10], DUBD [18], BM3D [9], DIDN [16], DnCNN [13], LIGN [23], VDIR [22]. Tables 1, 2 and 3 list the PSNR and SSIM obtained for all models in the SIDD, DND, and PolyU datasets, respectively, where higher metrics indicate superior denoising performance of the models.

Specifically, the results in Table 1 demonstrate that our MSIDNet achieves the highest PSNR and SSIM (i.e., 39.45 dB and 0.911) with the best denoising performance in the SIDD dataset, and that MSIDNet has 0.16 dB higher PSNR compared to the second ranking VDIR. In Table 2, our proposed MSIDNet achieves the second highest PSNR and SSIM values, and in Table 3, the highest PSNR is achieved, which shows that our method outperforms several other traditional methods and deep learning-based methods. Although COLA-Net performs fourth on the SIDD dataset, it slips to the middle of the rankings on both the DND and PolyU datasets. Combining the three table observations demonstrates that our MSIDNet achieves significant performance improvement over the three non-blind denoising models of BM3D, TWSC, and FFDNet, moreover, our method does not require to pre-set the noise level, which is more convenient for practical applications.

CBDNet uses additional data for training, compared to MSIDNet, which we only train with the SIDD dataset, our model performs better in denoising on entire datasets, specifically, our MSIDNet is 8.67 dB higher than CBDNet on the SIDD dataset, 1.66 dB higher than CBDNet on the DND dataset, and 0.94 dB higher than CBDNet on the PolyU dataset.

DIDN ranks fourth in performance on the DND dataset, but is second to last in performance on the SIDD dataset, with a PSNR of only 24.07 dB and a SSIM of only 0.350, and is also fourth to last in performance on the PolyU dataset. These results indicate that DIDN performs well only for

Table 1 Quantitative results on the SIDD dataset. The best and second-best are bold and underlined respectively

Methods	DnCNN [13]	DIDN [16]	BM3D [9]	DUBD [18]	TWSC [10]	FFDNet [14]	CBDNet [15]
PSNR	23.66	24.07	25.65	25.96	26.16	29.20	30.78
SSIM	0.583	0.350	0.685	0.442	0.483	0.594	0.744
Blind/non-blind	Blind	Blind	Non-blind	Blind	Non-blind	Non-blind	Blind
Methods	Path-Restore [20]	C2N [17]	HI-GAN [19]	COLA-Net [21]	LIGN [23]	VDIR [22]	MSIDNet (Ours)
PSNR	34.69	35.46	38.47	38.62	39.14	<u>39.29</u>	**39.45**
SSIM	0.869	0.878	0.900	0.900	0.907	<u>0.908</u>	**0.911**
Blind/non-blind	Blind	Blind	Blind	Blind	Blind	Blind	Blind

Table 2 Quantitative results on the DND dataset. The best and second-best are bold and underlined respectively

Methods	DnCNN [13]	BM3D [9]	C2N [17]	FFDNet [14]	TWSC [10]	CBDNet [15]	Path-Restore [20]
PSNR	32.43	34.51	37.28	37.61	37.94	38.06	39.00
SSIM	0.790	0.850	0.923	0.941	0.940	0.942	0.953
Blind/non-blind	Blind	Non-blind	Blind	Non-blind	Non-blind	Blind	Blind
Methods	COLA-Net [21]	HI-GAN [19]	DUBD [18]	DIDN [16]	VDIR [22]	LIGN [23]	MSIDNet (Ours)
PSNR	39.07	39.32	39.37	39.62	39.63	**39.90**	<u>39.72</u>
SSIM	0.949	0.952	0.952	0.953	0.953	**0.959**	<u>0.954</u>
Blind/non-blind	Blind	Blind	Blind	Blind	Blind	Blind	Blind

Table 3 Quantitative results on the PolyU dataset. The best and second-best are bold and underlined respectively

Methods	Path-Restore [20]	DnCNN [13]	BM3D [9]	DIDN [16]	TWSC [10]	CBDNet [15]	COLA-Net [21]
PSNR	31.73	34.29	35.44	36.00	36.09	36.29	36.33
SSIM	0.914	0.912	0.940	0.911	0.917	**0.954**	0.944
Blind/non-blind	Blind	Blind	Non-blind	Blind	Non-blind	Blind	Blind
Methods	FFDNet [14]	LIGN [23]	HI-GAN [19]	VDIR [22]	C2N [17]	DUBD [18]	MSIDNet (Ours)
PSNR	36.42	36.60	36.76	36.84	36.95	<u>37.01</u>	**37.23**
SSIM	0.933	0.924	<u>0.951</u>	0.934	0.950	0.941	0.947
Blind/non-blind	Non-blind	Blind	Blind	Blind	Blind	Blind	Blind

denoising a particular dataset and has poor generalization ability. The recent method LIGN is in the high performance for both SIDD and DND datasets, but the performance drops dramatically to the middle when dealing with PolyU dataset, which indicates that although LIGN is effective in dealing with most datasets, it performs poorly in denoising when dealing with a particular dataset. Such phenomenon is also seen in Path-Restore, C2N, VDIR, and FFDNet, which have unstable denoising performance, in comparison to our MSIDNet is performs well.

The quantitative experimental results demonstrate that our proposed MSIDNet performs optimally compared to thirteen other image denoising methods.

4.4 Qualitative comparison

As shown in Figs. 4, 5, 6, and 7, the qualitative comparison results on the SIDD, DND, PolyU, and RNI15 datasets are presented sequentially to demonstrate the superior performance of our proposed MSIDNet. As can be seen in Fig. 4, the numbers and letters on the wooden blocks become blurred and the edge contours are lost due to the noise, resulting in a very poor visual effect of the image. Among these methods, DIDN, DUBD, TWSC, FFDNet, BM3D, and DnCNN do not restore the image effectively, which not only leaves a lot of noise in the image but also creates artifacts. The remaining methods remove noise from the image but also cause loss of image detail and edge information, of which CBDNet is the most severe with the figures on the wooden blocks being the least sharp. Our MSIDNet results are the most similar to the ground-truth images, producing the clearest images.

Figure 5 shows the qualitative comparison results for the stone pillar images from the DND dataset. The carving on the surface of the stone column is uneven, making it very difficult to remove the noise from the image. Most of the methods do not effectively remove the noise from the images and perform poorly. Among them, TWSC loses most of the sculpted shapes, leaving many over-smoothing regions in the image. C2N and DnCNN not only lose the sculptural texture, but also generate artifacts in the image, which makes the

image more blurred. Compared to other methods, our proposed model can retain more details of stelae carvings while removing noise, and thus has the most outstanding visual effect. Figure 6 presents the results of the qualitative comparison of the leaf images on the PolyU dataset. As can be seen from the denoising results, Path-Restore and CBDNet cause the color of the wall behind the leaf to change, which makes the image content inaccurate. DnCNN and FFDNet generate severe over-smoothing phenomena while denoising. Although the PSNR of C2N is close to our MSIDNet, C2N causes the texture details and edge information of the leaves to be lost, while our MSIDNet closely resembles the original leaf image with excellent denoising performance. Figure 7 lists the qualitative comparison of three different scenarios from the RNI15 dataset. By looking at the eyes of the dog, the border of the window, and the pillar of the lamp, we can see that BM3D produces artifacts that severely affect the image and cause the image quality to degrade. Although DIDN and COLA-Net remove most of the noise, there are still remaining noise influences and image details are lost. Since the RNI15 dataset is without ground-truth images, we introduce the natural image quality evaluator (NIQE) [41] to evaluate the image quality. A smaller NIQE value indicates a better overall naturalness of the image. Our method has the smallest NIQE in all scenes. The results demonstrate that our method performs optimally in all scenes.

The characteristics of the image noise in the SIDD, DND, RNI15, and PolyU datasets are different depending on the capturing device and method. We use only the SIDD dataset to train the proposed MSIDNet, which performs well on all four datasets, demonstrating the excellent generalization of our model.

4.5 Ablation study

To demonstrate that the proposed RAB and FEB are effective, ablation studies are conducted for each of these two blocks. The SIDD dataset and model parameters for the ablation study are the same as in the previous experiments. Table 4 lists the results obtained by replacing the RAB with the residual block [26] and the dense block

Fig. 4 Qualitative comparison of image denoising results from the SIDD dataset
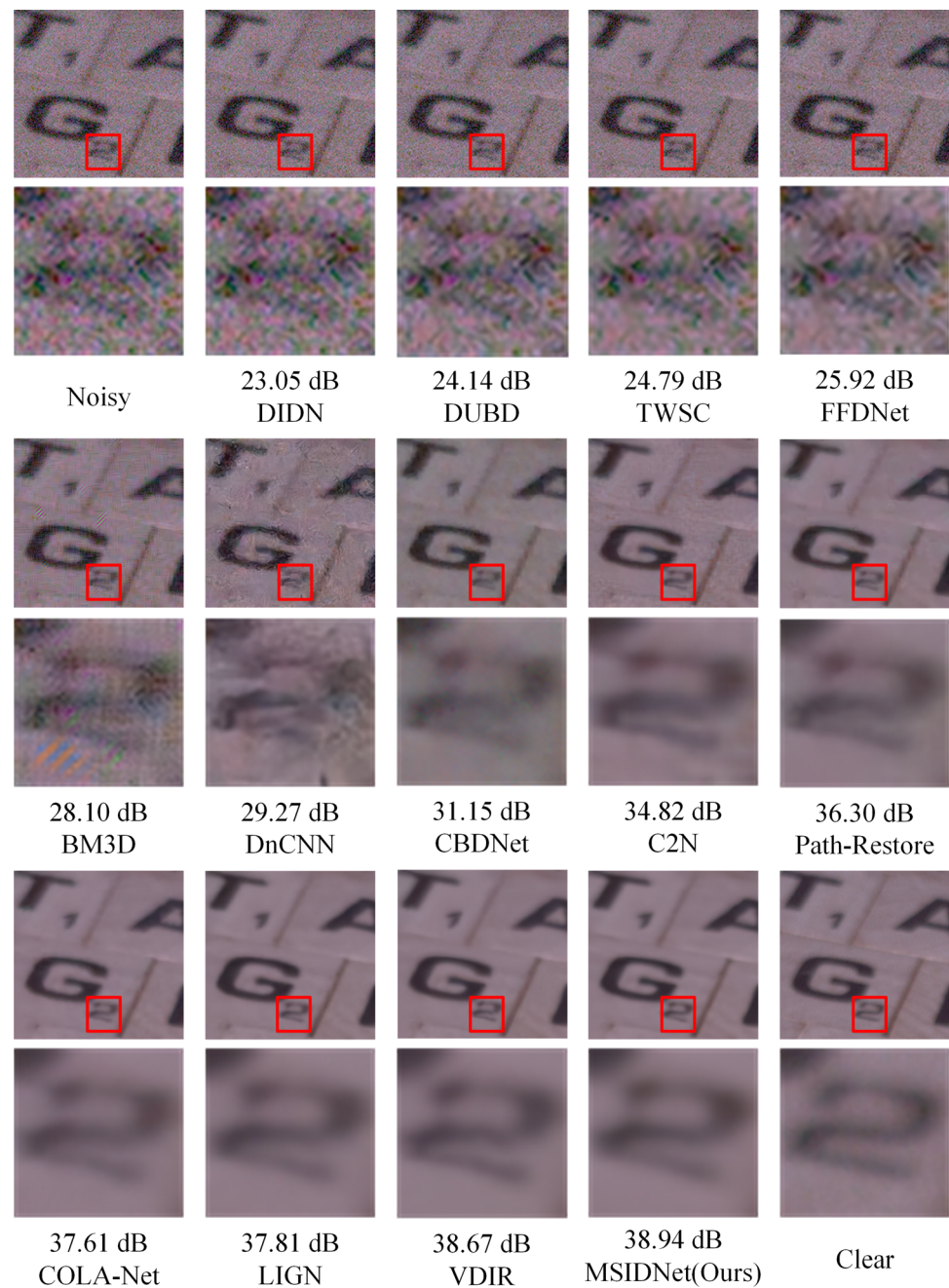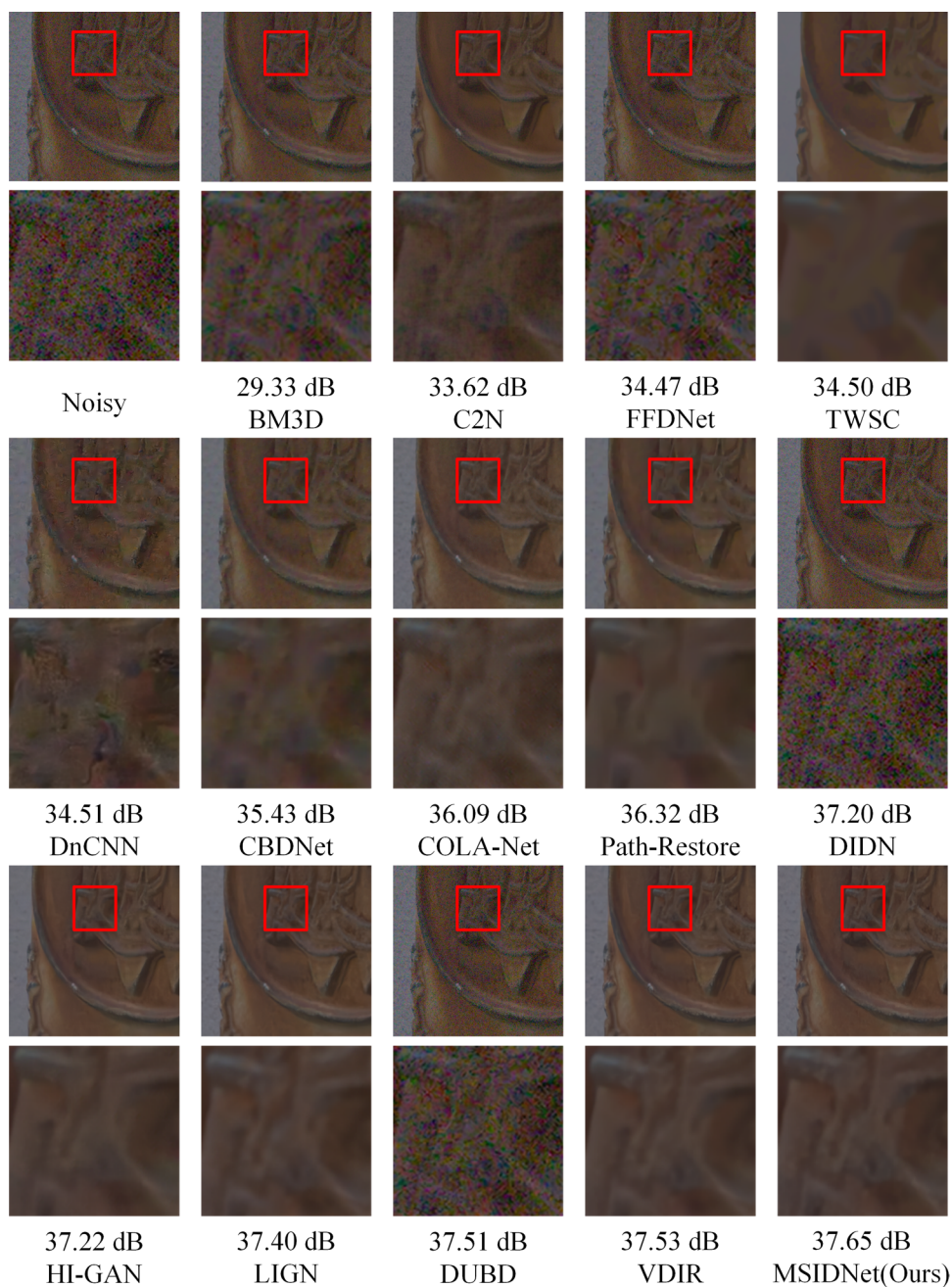

[42], respectively. As can be seen from the results, compared to RAB, the PSNR of the residual block decreases by 0.34 dB and the SSIM decreases by 0.005, the PSNR of the dense block decreases by 0.4 dB and the SSIM decreases by 0.004, demonstrating the optimal performance of our block. Table 5 shows that without FEB, the PSNR decreases by 0.28 dB and the SSIM decreases by 0.004, and the experimental results demonstrate that our designed blocks are effective.

4.6 Limitations

Real-world noise is often superimposed on multiple types, and it is challenging for the network to restore clean images in blind denoising. The results of the comparison experiments illustrate that the performance of our proposed network needs further improvement. Optimization of the network structure is beneficial for improving the feature extraction ability and enhancing the discrimination between

Fig. 5 Qualitative comparison of image denoising results from the DND dataset

| | 29.33 dB | 33.62 dB | 34.47 dB | 34.50 dB |
| Noisy | BM3D | C2N | FFDNet | TWSC |

| 34.51 dB | 35.43 dB | 36.09 dB | 36.32 dB | 37.20 dB |
| DnCNN | CBDNet | COLA-Net | Path-Restore | DIDN |

| 37.22 dB | 37.40 dB | 37.51 dB | 37.53 dB | 37.65 dB |
| HI-GAN | LIGN | DUBD | VDIR | MSIDNet(Ours) |

noise and texture information. Using the latest data enhancement and transfer learning techniques to improve robustness of the network can make better use of the limited real-world dataset.

4.7 Application for object detection

In this section, we apply the denoised images to the object detection task and let the different methods be further compared. Since there is no object detection dataset containing real-world noise, we use the PolyU dataset for labeling. There are three classes: person, lock, and leaf. We use the denoised images of different methods to train YOLOv5 [43] respectively, and the results are shown in Table 6. Different experimental results were obtained using denoised images from various methods with the same settings. From the results, we can see that the denoised images are more beneficial for object detection than the noisy images, and the metrics are improved. Our MSIDNet achieves the highest in precision, mAP@0.5, and mAP@0.5:0.95 metrics. Experimental results demonstrate that our method preserves the image information well after denoising and performs well when applied to object detection.

Fig. 6 Qualitative comparison of image denoising results from the PolyU dataset

5 Conclusion

We propose a multi-scale image denoising network for removing real-world noise, named MSIDNet. Multi-scale features between layers are fully exploited by fusion mechanisms that enhance network context-awareness. The FEB further complements feature information by fusing global and local features to exploit their complementary nature. Qualitative and quantitative comparison experiments demonstrate that MSIDNet performs well. The denoised image preserves more edge details and improves the over-smoothing problem, which enhances the visual effect. In the future, we will deepen our research on real-world noise and further improve the network denoising effect.

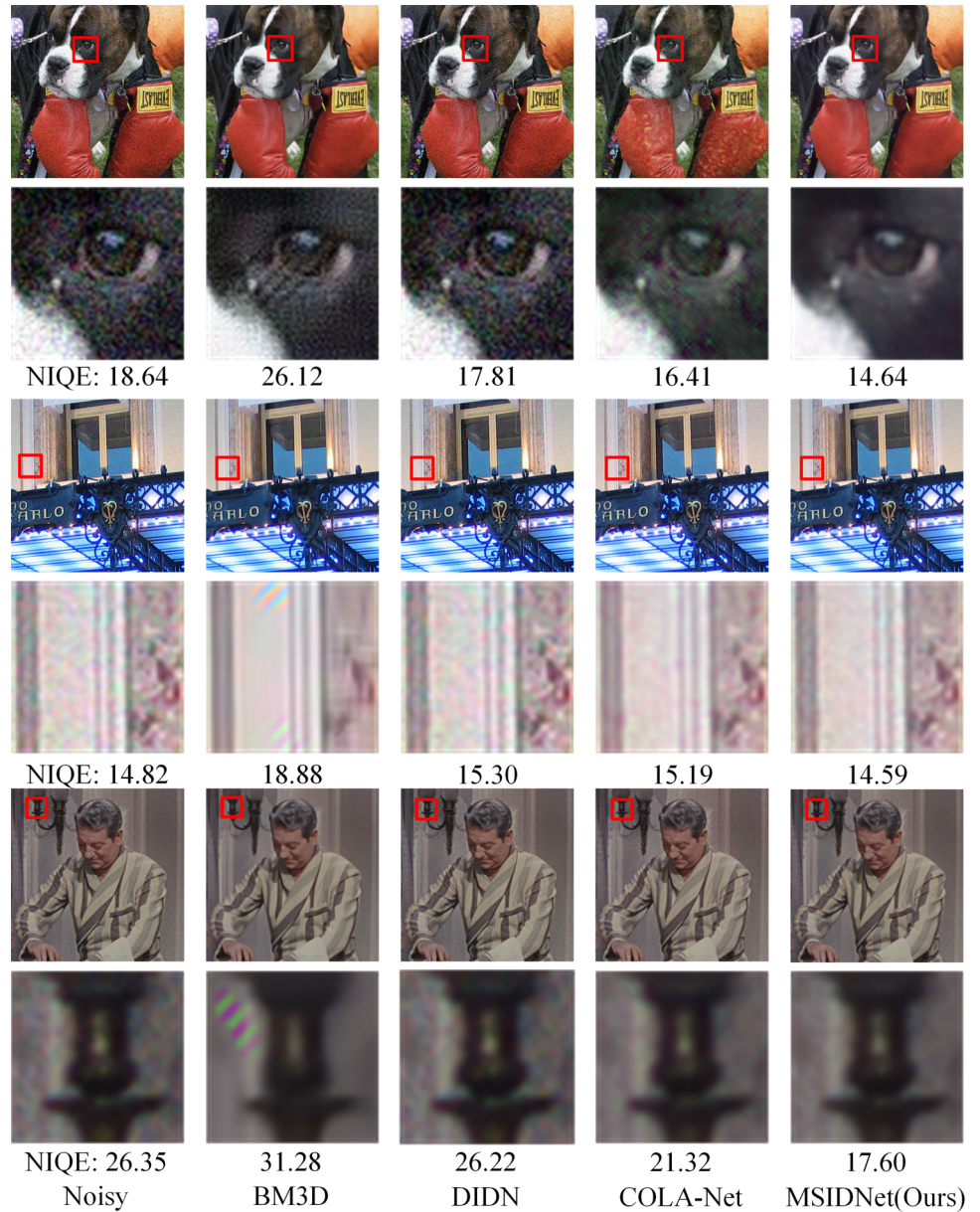**Fig. 7** Qualitative comparison of image denoising results from the RNI15 dataset

NIQE: 18.64	26.12	17.81	16.41	14.64
NIQE: 14.82	18.88	15.30	15.19	14.59
NIQE: 26.35	31.28	26.22	21.32	17.60
Noisy	BM3D	DIDN	COLA-Net	MSIDNet(Ours)

Table 4 RAB vs residual block and dense block

Methods	Dense Block	Residual Block	RAB
PSNR	39.05	39.11	39.45
SSIM	0.907	0.906	0.911

Table 5 Ablation of FEB

Methods	Without FEB	FEB
PSNR	39.17	39.45
SSIM	0.907	0.911

Table 6 Comparison of object detection results. The best and second-best are bold and underlined respectively

Methods	Precision	Recall	mAP@.5	mAP@.5:.95
Noisy	0.408	0.524	0.524	0.282
BM3D [9]	0.435	0.565	0.590	0.302
DIDN [16]	0.454	0.665	0.557	0.308
C2N [17]	0.545	0.578	0.650	0.338
COLA-Net [21]	0.595	0.618	0.642	0.326
DUBD [18]	0.605	**0.667**	0.634	0.336
MSIDNet (ours)	**0.669**	0.543	**0.651**	**0.343**

Declarations

Conflict of interest The authors declare that they have no conflict of interest.

References

1. Meng L, Ding S, Xue Y (2017) Research on denoising sparse autoencoder. Int J Mach Learn Cybern 8(5):1719–1729
2. Cao F, Gao C, Ye H (2022) A novel method for image segmentation: two-stage decoding network with boundary attention. Int J Mach Learn Cybern 13(5):1461–1473
3. Lian G, Wang Y, Qin H, Chen G (2022) Towards unified on-road object detection and depth estimation from a single image. Int J Mach Learn Cybern 13(5):1231–1241
4. Raveendran AP, Alzubi JA, Sekaran R, Ramachandran M (2021) A high performance scalable fuzzy based modified Asymmetric Heterogene Multiprocessor System on Chip (AHt-MPSOC) reconfigurable architecture. J Intell Fuzzy Syst 42(2):647–658
5. Movassagh AA, Alzubi JA, Gheisari M, Rahimi M, Mohan S, Abbasi AA, Nabipour N (2021) Artificial neural networks training algorithm integrating invasive weed optimization with differential evolutionary model. J Ambient Intell Hum Comput. https://doi.org/10.1007/s12652-020-02623-6
6. Babu MV, Alzubi JA, Sekaran R, Patan R, Ramachandran M, Gupta D (2021) An improved IDAF-FIT clustering based ASLPP-RR routing with secure data aggregation in wireless sensor network. Mobile Netw Appl 26(3):1059–1067
7. Alrabea A, Alzubi O, Alzubi J (2020) An enhanced Mac protocol design prolong sensor network lifetime. IRECAP 10(1):37–43
8. Buades A, Coll B, Morel J M (2005) A non-local algorithm for image denoising. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp 60–65
9. 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
10. Xu J, Zhang L, Zhang D (2018) A trilateral weighted sparse coding scheme for real-world image denoising. In: Proceedings of the European Conference on Computer Vision (ECCV), pp 20–36
11. Jain V, Murray J F, Roth F, Turaga S, Zhigulin, V, Briggman K L, Seung H S (2007) Supervised learning of image restoration with convolutional networks. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), pp 1–8
12. Burger H C, Schuler C J, Harmeling S (2012) Image denoising: Can plain neural networks compete with BM3D? In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp 2392–2399
13. 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
14. Zhang K, Zuo W, Zhang L (2018) FFDNet: Toward a fast and flexible solution for CNN-based image denoising. IEEE Trans Image Process 27(9):4608–4622
15. Guo S, Yan Z, Zhang K, Zuo W, Zhang L (2019) Toward convolutional blind denoising of real photographs. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp 1712–1722
16. Yu S, Park B, Jeong J (2019) Deep iterative down-up cnn for image denoising. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)
17. Jang G, Lee W, Son S, Lee K M (2021) C2N: Practical generative noise modeling for real-world denoising. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), pp 2350–2359
18. Soh J W, Cho N I (2021) Deep universal blind image denoising. In: International Conference on Pattern Recognition (ICPR), pp 747–754
19. Vo DM, Nguyen DM, Le TP, Lee SW (2021) HI-GAN: a hierarchical generative adversarial network for blind denoising of real photographs. J Autom Inf Sci 570:225–240
20. Yu K, Wang X, Dong C, Tang X, Loy CC (2021) Path-restore: learning network path selection for image restoration. IEEE Trans Pattern Anal Mach Intell 44:7078–7092
21. Mou C, Zhang J, Fan X, Liu H, Wang R (2021) COLA-Net: collaborative attention network for image restoration. IEEE Trans Multimedia 24:1366–1377
22. Soh JW, Cho NI (2022) Variational deep image restoration. IEEE Trans Image Process 31:4363–4376
23. Qiao S, Yang J, Zhang T, Zhao C (2022) Layered input GradiNet for image denoising. KNOWL-BASED SYST 254:109587
24. Misra D, Nalamada T, Arasanipalai A U, Hou Q (2021) Rotate to attend: Convolutional triplet attention module. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACVW), pp 3139–3148
25. Zamir S W, Arora A, Khan S, Hayat M, Khan F S, Yang M H (2021) Restormer: efficient transformer for high-resolution image restoration, arXiv preprint https://arxiv.org/abs/2111.09881
26. He K, Zhang X, Ren S, Sun J (2016) Deep residual learning for image recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp 770–778
27. Zamir S W, Arora A, Khan S, Hayat M, Khan F S, Yang M H, Shao L (2020) Learning enriched features for real image restoration and enhancement. In: European Conference on Computer Vision (ECCV), pp 492–511
28. Abdelhamed A, Lin S, Brown M S (2018) A high-quality denoising dataset for smartphone cameras. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp 1692–1700
29. Plotz T, Roth S (2017) Benchmarking denoising algorithms with real photographs. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp 1586–1595
30. Xu J, Li H, Liang Z, Zhang D, Zhang L (2018) Real-world noisy image denoising: a new benchmark, arXiv preprint https://arxiv.org/abs/1804.02603
31. Lebrun M, Colom M, Morel JM (2015) The noise clinic: a blind image denoising algorithm. Image Process Line 5:1–54
32. Shi W, Caballero J, Huszár F, Totz J, Aitken A P, Bishop R, Wang Z (2016) Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp 1874–1883
33. Chen X, Sun C (2022) Multiscale recursive feedback network for image super-resolution. IEEE Access 10:6393–6406
34. Yan X, Tang H, Sun S, Ma H, Kong D, Xie X (2022) After-unet: axial fusion transformer unet for medical image segmentation. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WCACV), pp 3971–3981
35. Zhou F, Sun X, Dong J, Zhao H, Zhu X X (2021) SurroundNet: towards effective low-light image enhancement, arXiv preprint https://arxiv.org/abs/2110.05098

36. Sun Z, Zhou W, Ding C, Xia M (2022) Multi-resolution transformer network for building and road segmentation of remote sensing image. ISPRS Int J Geo-Inf 11(3):165

37. Liu Z, Lin Y, Cao Y, Hu H, Wei Y, Zhang Z, Guo B (2021) Swin transformer: hierarchical vision transformer using shifted windows. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), pp 10012–10022

38. Huang S, Lu Z, Cheng R, He C (2021) FaPN: Feature-aligned pyramid network for dense image prediction. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), pp 864–873

39. Charbonnier P, Blanc-Feraud L, Aubert G, Barlaud M (1994) Two deterministic half-quadratic regularization algorithms for computed imaging. In: Proceedings of International Conference on Image Processing (ICIP), pp 168–172

40. Setiadi DRIM (2021) PSNR vs SSIM: imperceptibility quality assessment for image steganography. Multimed Tools Appl 80(6):8423–8444

41. Mittal A, Soundararajan R, Bovik AC (2012) Making a "completely blind" image quality analyzer. IEEE Signal Process Lett 20(3):209–212

42. Huang G, Liu Z, Van Der Maaten L, Weinberger K Q (2017) Densely connected convolutional networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp 4700–4708

43. Jocher G, Stoken A, Borovec J, Chaurasia A, Changyu L, Laughing A V, Ingham F (2021) ultralytics/yolov5: v5. 0-YOLOv5-P6 1280 models AWS Supervise. ly and YouTube integrations. Zenodo 11