 DOI: 10.35382/TVUJS.14.8.2024.80

THE PROBLEM OF IMAGE SUPER-RESOLUTION, DENOISING AND IMAGE RESTORATION METHODS IN DEEP LEARNING MODELS

Ngoc-Giau Pham[1,6], Thanh-Hai Tong Le[2*], Van-Hieu Duong[3], Hong-Ngoc Tran[4], Phuoc-Hung Vo[5]

Abstract – *This article addresses the challenges of image super-resolution and noise reduction, which are crucial for enhancing the quality of images derived from low-resolution or noisy data. Several approaches were compared and assessed for upgrading low-resolution images to higher resolutions and eliminating unwanted noise while maintaining the essential characteristics of the original images and recovering images from poor quality or damaged data using deep learning models. It is indicated by research analysis and the experimental outcomes on image quality metrics that the ED-Unet neural network model, enhanced with pretrained weights, significantly outperforms other methods, achieving a Train PSNR of 30.791, a Valid PSNR of 30.699, and a Test PSNR of 31.0172.*

Keywords: denoising, ED-Unet, image restoration, pretrained weights, resolution.

I. INTRODUCTION

In the increasingly digital world, images and videos are essential in fields ranging from education and healthcare to entertainment and security. However, the quality of these visual media often falls short of expectations due to limitations in camera technology, poor lighting conditions, or errors during transmission. This situation has spurred a high demand for image quality enhancement techniques such as super-resolution, noise reduction, and image restoration. These methods not only improve the clarity and quality of images but also enable the recovery of valuable information that was previously obscured by low resolution or noise [1].

In recent years, deep learning has become a powerful tool in addressing various image processing challenges, including super-resolution, denoising, and image restoration [2]. The advancement of deep neural networks and convolutional neural networks (CNNs) has led to significant progress, enabling higher precision in image handling, automating restoration processes, and reducing the time required for processing (Figure 1).

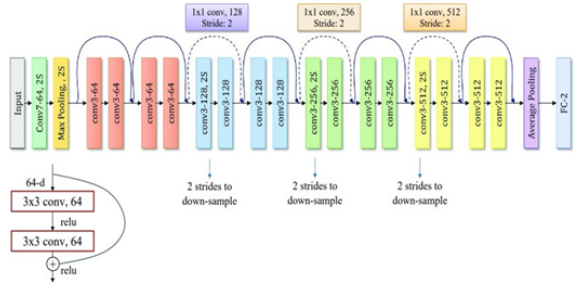

Fig. 1: Deep residual learning architecture [2]

Recent advancements in image super-resolution and noise reduction are driven primarily by the application of deep learning techniques. Wang et al. [3] provide a comprehensive survey on various deep learning approaches to image super-resolution, detailing their effectiveness in enhancing low-resolution images with a focus on the progression of CNNs and residual learning techniques. Additionally, Liang et al. [4] introduce adaptive CNNs for

[1,2,3]Tien Giang University, Vietnam
[4]Viet Duc University, Vietnam
[5]Tra Vinh University, Vietnam
[6]Postgraduate student, Tra Vinh University, Vietnam
*Corresponding author: tonglethanhhai@tgu.edu.vn

Received date: 02nd July 2024; Revised date: 07th August 2024; Accepted date: 16th September 2024

noise reduction in low-light images, showcasing how dynamic adaptation to noise levels can significantly enhance image clarity.

Furthermore, Hüsem et al. [5] and Moghimi et al. [6] explore the capabilities of generative adversarial networks (GANs) and optimized deep convolutional networks for real-time super-resolution applications, respectively. These studies highlight the potential of GANs in reconstructing high-quality images from degraded inputs and the feasibility of deploying deep learning models in real-time scenarios without quality compromise.

Table 1: Comparison of the design of the most relevant state-of-the-art deep learning architectures for image super-resolution

Model name	Publication	Upsampling	Design keywords	Loss function	Training dataset	PSNR
SRCNN [7]	ECCV 2014	Bicubic	CNN	L_2	DIV2k	27.50
DRCN [8]	CVPR 2016	Bicubic	Residual, Recursive	L_2	DIV2k	28.02
FSRCNN [9]	ECCV 2016	Deconvolution	Residual	L_2	General100+ T91	27.59
ESPCN [10]	CVPR 2017	Sub-Pixel	Residual	L_2	DIV2k	27.73
SRResNet [11]	CVPR 2017	Sub-Pixel	Residual	L_2	DIV2k	28.49
SRGAN [11]	CVPR 2017	Sub-Pixel	Residual	$L_{content}$, $L_{Adversarial}$	DIV2k	26.02
EDSR [12]	CVPRW 2017	Sub-Pixel	Residual	L_1	DIV2k	28.80
EnhanceNet [13]	ICCV 2017	Bicubic	Residual	$L_{content}$, $L_{Adversarial}$, $L_{Texture}$	DIV2k	25.77
MemNet [14]	ICCV 2017	Bicubic	Residual, Recursive, Dense	L_2	DIV2k	28.26
SRDenseNet [15]	ICCV 2017	Deconvolution	Residual, Dense	L_2	ImageNet(subnet)	28.50
RDN [16]	CVPR 2017	Sub-Pixel	Residual, Dense	L_1	DIV2k	28.81
CARN [17]	ECCV 2017	Sub-Pixel	Recursive, Residual, Dense	L_1	DIV2k	28.60
RCAN [18]	ECCV 2017	Sub-Pixel	Residual, Attention	L_1	DIV2k	28.87
ESRGAN [19]	ECCVW 2017	Sub-Pixel	Residual, Dense	L_1, $L_{Texture}$ $L_{content}$	DIV2k+Flickr2k	-
RNAN [20]	ICLR 2019	Sub-Pixel	Residual, Attention	L_1	DIV2k	28.83
SAN [21]	CVPR 2019	Sub-Pixel	Residual, Attention	L_1	DIV2k	28.92
HAN [22]	ECCV 2020	Sub-Pixel	Residual, Attention	L_1	DIV2k	28.90
IPT [23]	CVPR 2021	Sub-Pixel	Transformers	L_1	ImageNet	29.01
SwinIR [24]	ICCV 2021	Sub-Pixel	Transformers	L_1	DIV2k	28.94
DiffIR [25]	ICCV 2023	Sub-Pixel	Transformers	L_1	DIV2k	29.13

This paper aims to explore and analyze advanced methods in the fields of super-resolution, denoising, and image restoration using deep learning models. This research delves into the theoretical foundations of these models, assesses their performance through case studies, and discusses how these techniques can be optimized and implemented in real-world scenarios. Additionally, the paper highlights the challenges and opportunities in developing and deploying deep learning solutions for image processing, aiming to achieve the best possible outcomes in today's digital age.

The structure of this paper is organized as follows: the second section presents methods for image restoration and noise reduction, and the third section compares the performance of these methods with machine learning algorithms based on the results obtained from experiments.

II. METHODOLOGY

A. Problem statement

Image super-resolution is the process of enhancing the resolution of images by using machine learning algorithms to predict and reconstruct image details at a higher pixel level. In deep learning, models such as CNNs are trained to recognize and reproduce the subtle features of images from large datasets [25]. The goal is to sharpen images from low-resolution cameras, making them suitable for applications such as healthcare for more accurate diagnostics or in surveillance to enhance clarity in identification.

Image denoising is the process of removing noise that may arise from poor lighting conditions or sensor errors in cameras. Deep learning models such as denoising autoencoders or CNNs are trained to distinguish between noise and useful signals in images. By training on datasets with known noise, the model gradually learns to restore the original image by minimizing the presence of noise without losing essential image details (Table 1) [26].

Image restoration involves techniques for reinstating damaged or missing portions of an image. This is commonly seen in the restoration of ancient artworks, historical photographs, or enhancing images taken with low-quality devices. Networks such as convolutional neural networks are trained to predict and reconstruct missing parts based on surrounding image patterns and previously acquired knowledge [27]. This process requires high precision and a deep understanding of the nature of images and the structure of the surrounding environment.

While these techniques offer numerous benefits, they also face challenges such as high

computational demands and the ability to generalize across data that differs from the training set. However, with the continuous advancement of computer technology and deep learning algorithms, the potential for improvements and applications in image super-resolution, denoising, and restoration is immense [28]. This opens up many opportunities in fields such as security, healthcare, and entertainment.

B. Concepts

The research direction presented in this paper focuses on developing a network capable of performing image denoising and super-resolution, referred to as the enhanced denoising convolutional neural network. This network utilizes the residual learning approach of convolutional networks and introduces a pretrained weights methodology. Rather than learning end-to-end mapping, the network is trained to generate a new image. This new image represents the difference between the input image and the original (template) image. The aim of this system is to eliminate noise from the image and subsequently enhance the image resolution (super-resolution) (Figure 2) [29].

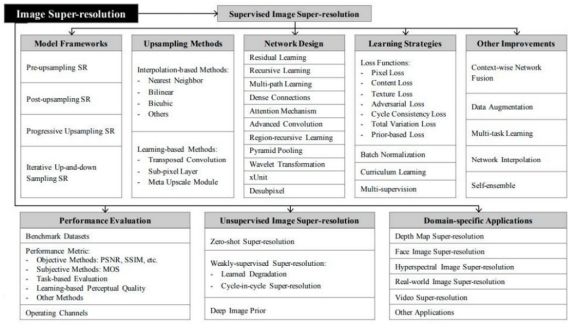

Fig. 2: Base Deep Learning Method [27]

Skip connection

Degradation problem: In theory, deep networks can learn complex functions and are more effective than shallow networks. However, in practice, the performance of the network degrades as more layers are added. Although part of this problem could be attributed to the vanishing gradient, the authors of ResNet have addressed this by initializing weights and employing batch normalization to ensure a healthy gradient flow. Despite these measures, performance still decreases [1].

Skip connection: The output of one layer is skipped over several subsequent layers and reintegrated as input to another layer, which is positioned further along, beyond multiple intermediate layers. Skip connections can vary depending on the architecture in which they are implemented. The ResNet architecture utilizes skip connections to address the degradation problem, while the DenseNet architecture also incorporates this feature but arranges the connections in a distinct manner to improve information and gradient flow throughout the network.

Neural networks are capable of learning complex, high-dimensional functions, provided that these functions are nonconvex.

Visualizations of the loss surface indicate that skip connections facilitate smoother learning and faster convergence (Figure 3).

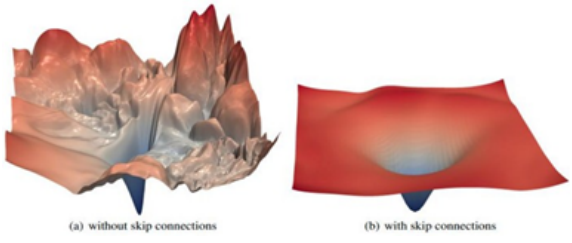

Fig. 3: Complex loss surface [30]

Alternatives to increase feature resolution

During the super-resolution process, the image should be upsampled at some point in the network. There are different strategies regarding both when and how this is done. Depending on the combination chosen, the performance will vary accordingly.

a) Interpolation (Figure 4): This is a commonly employed technique for enlarging images by calculating the values of new pixels based on the neighboring pixels. Popular interpolation methods include nearest neighbor interpolation, bilinear interpolation, and bicubic interpolation.

Each method has its own advantages and disadvantages in preserving image quality during resizing.

b) Reshape : This method alters the shape of a data array without modifying the data within it. In image processing, this can involve changing how pixels are arranged in memory, thereby altering the size of the image. However, it is not the primary method for enhancing image resolution but is commonly used in adjusting the data structure.

c) Transposed Convolution (Figure 5): This is a type of transposed convolutional layer, used in neural networks to increase the size of feature maps. It operates by inserting spaces between pixels, and then applying convolution to fill these gaps, effectively increasing the resolution of the output [29].

d) Pixel Shuffle (Figure 6): This is a technique used in deep learning models, particularly in tasks involving super-resolution. It restructures the channels of feature maps into pixel positions in the image space, commonly employed to create resolution without losing information. This method effectively enhances image quality by redistributing the feature data across the spatial dimensions of the image.

e) Unpooling (Figure 7): During the pooling process, positional information about pixels is discarded to reduce the size of the feature map. Unpooling is the reverse process, where positional information is restored and the size of the feature map is increased. This is an important element in some neural network architectures to enhance the detail of images or feature maps during the upsampling process.

Through implementation and experimentation on The Oxford- IIIT Pet dataset [31], which includes 37 categories of pets with approximately 200 images per class, characterized by large variations in scale, pose, and lighting, and all images having associated ground truth annotations of breed, head ROI, and pixel-level trimap segmentation. Based on the results presented in Table II, the interpolation (bilinear) method is the most effective among the tested techniques. This outcome is based on the evaluation of test loss,

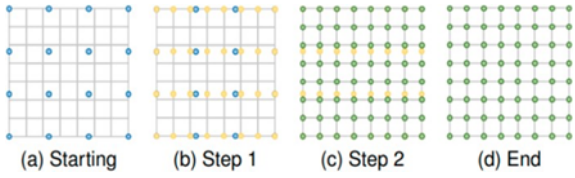

Fig. 4: Interpolation-based upsampling methods: the grey board denotes the coordinates of pixel (a), and the blue, yellow, and green points represent the initial (b), intermediate (c), and output pixels respectively (d) [27]

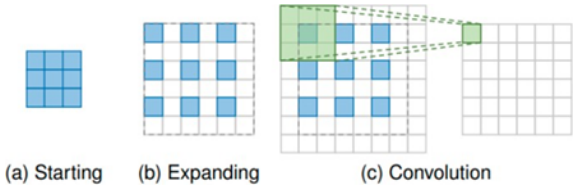

Fig. 5: Transposed Convolution or Deconvolution layer: the blue boxes denote the input (a), and the green boxes indicate the kernel (b) and the convolution output (c) [27]

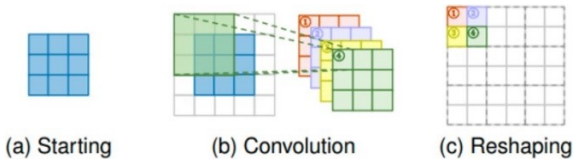

Fig. 6: Pixel Shuffle layer: the blue boxes denote the input (a), and the boxes with other colours indicate different convolution operations (b) and different output feature maps (c) [27]

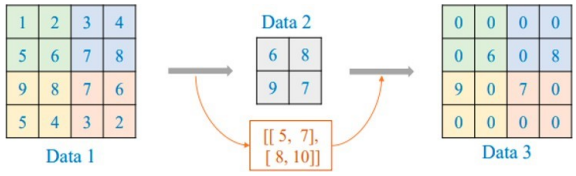

Fig. 7: Unpooling layer: Simply replacing each entry of a feature map by an s x s block with the entry value in the top left corner zeroing elsewhere and making use of max-pooling indices computed in the max-pooling layers of the convolutional subnetwork [27]

with a value of 0.4025, which is lower than the other methods. This indicates that the bilinear interpolation method has significantly reduced noise and improved image quality more than other methods such as Reshape, Pixel Shuffle, Unpooling, and Transposed Convolution.

Table 2: Comparison of some of the upsampling methods (epochs = 30)

Upsampling	Loss function	Model name	Train loss	Test loss	Test image
Interpolation (bilinear)	Cross Entropy Loss()	ED-Unet	0.1522	**0.4025**	80
Reshape	Cross Entropy Loss()	ED-Unet	0.2455	0.6648	80
Pixel Shuffle	Cross Entropy Loss()	ED-Unet	0.1154	0.4561	80
Unpooling	Cross Entropy Loss()	ED-Unet	0.0991	0.5466	80
Convolution Transpose	Cross Entropy Loss()	ED-Unet	0.1296	0.4788	80

The proposed network architecture and the pretrained weights method

The proposed ED-Unet model (Figure 8) is a 23-layer deep convolutional neural network consisting of two layers processing the input image, four encoder blocks (each with two layers), four decoder blocks (each with three layers), and one layer processing the output image. The activation function used is Tanh, and the loss function employed is mean squared error (MSE). The kernel matrix size is 3 x 3, uniform across all 23 convolutional layers. The first convolutional layer receives 3 feature maps corresponding to the Red Green Blue (RGB) channels of the image, while layers 2 through 22 each receive 64 feature maps. The final output of the network has three channels. All weights are initialized from a standard random distribution. With such architecture, the total number of parameters in the proposed network is 28,985,088. During the training phase, the network input is a 256 x 256 two-dimensional interpolated image, and the target output is a new 256 x 256 image. The network is capable of operating with any input image size, utilizing the weights it has learned during training. The number of trainable parameters for the convolutional layers does not depend on the input size.

From the analyses presented, the paper proposes a network architecture comprising Fblock units for preprocessing and information extraction, Eblock units for denoising and image quality enhancement, and Dblock units for output restoration. This structured approach allows for sequential improvements in image processing, culminating in enhanced and restored final outputs.

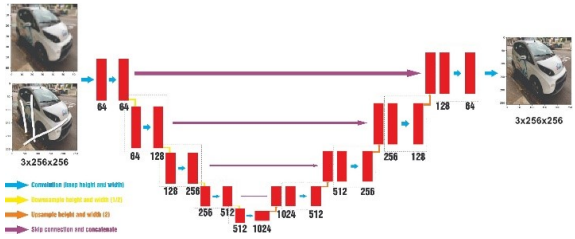

Fig. 8: The proposed network architecture

The method of pretrained weights, also known as the use of pre-trained model parameters, involves utilizing model parameters that have already been trained on a robust dataset. This approach enhances the learning speed and performance of the model on a new dataset, particularly when the available data for a specific task is noisy, incomplete, or less informative compared to the original dataset.

Pixel loss

The objective of the loss function is to measure the difference between two images on a pixel-by-pixel basis to allow the images to converge as closely as possible. By definition, pixel loss is highly correlated with the peak signal-to-noise ratio (PSNR), which has led to its rapid adoption as the most favored loss function [32, 33].

The L1 loss function can be expressed as Formula (1).

$$L1(I_{SR}, I_y) = \frac{1}{hwc}\sum_{i,j,k}\left|I_{SR}^{i,j,k} - I_y^{i,j,k}\right| \quad (1)$$

The Mean Square Error (MSE) loss function

can be expressed as Formula (2).

$$L_{MSE}(I_{SR}, I_y) = \frac{1}{hwc}\sum_{i,j,k}\left(I_{SR}^{i,j,k} - I_y^{i,j,k}\right)^2 \quad (2)$$

Wherein:

h: height of the image

w: width of the image

c: channel of the image

$I_y \in \mathbb{R}^{h\times w}$: ground-truth high-resolution image

$I_{SR} \in \mathbb{R}^{h\times w}$: super-resolution image

Evaluation metrics: PSNR is one of the most popular metrics not only for image super-resolution but also for lossy compression, image inpainting, or image denoising. Therefore, it is often used as the main performance measurement for most benchmarks. This measurement provides an objective assessment of the reconstruction error at the pixel level. The MSE is defined as Formula (3).

$$MSE = \frac{1}{HW}\sum_{i=0}^{H-1}\sum_{j=0}^{W-1}\left(I_y(i,j) - I_{SR}(i,j)\right)2 \quad (3)$$

The PSNR is defined as Formula (4).

$$PSNR = 10.\log_{10}\left(\frac{MAX^2}{MSE}\right) \quad (4)$$

Wherein:

H: height of the image

W: width of the image

MAX: the maximum possible pixel of the image

$I_y \in \mathbb{R}^{h\times w}$: ground-truth high-resolution image

$I_{SR} \in \mathbb{R}^{h\times w}$: reconstructed image

Noise reduction and image quality enhancement unit

The noise reduction and image quality enhancement unit is a 3 x 3 kernel. The dimensions of the feature maps for layer i are represented as Di (i=1. . .6). The relationship between the convolutional layers can be expressed as follows in Formula (5).

$$D3 - D1 = D1 - D2 = d \quad (5)$$

In this context, d denotes the discrepancy between the first and second tiers, or between the first and third tiers.

Similarly, the dimensions of the channels in the submodule also exhibit this relationship and can be described as follows in Formula (6).

$$D_6 - D_4 = D_4 - D_5 = d \quad (6)$$

Assuming the input to this module is l_{k-1}, the output of the module is represented as Formula (7).

$$P_1^k = C_a(l_{k-1}) \quad (7)$$

Wherein, l_{k-1} denotes the output from the preceding block and serves as the input to the current block, and Ca represents the operation of sequential convolution.

The feature map matrices with dimensions $\frac{D_s}{S}$ and the inputs to the first convolutional layer are concatenated within the channel framework as Formula (8).

$$R^k = C\,(S(P_1^k, \tfrac{1}{s}), I_{k-1}) \quad (8)$$

Here, C and S represent the operations of concatenation and splitting, respectively. Specifically, the dimension of P_1^k is D3. Thus, S(P_1^k, $\frac{1}{s}$) indicates that features of dimension $\frac{D_s}{S}$ are retrieved from P_l^k. This operation concatenates these features with I_{k-1} within the channel framework. The goal is to integrate previous information with some current information. The remainder of the local short path information as the input for the kernel is used, primarily extracting further long-range feature mappings as Formula (9).

$$P_2^k = C_b(S\,(P_1^k, 1-\tfrac{1}{s})) \quad (9)$$

Here, P_2^k, and C_b correspond to the convolution operations of the output and the concatenation of the kernel. Therefore, the enhancement unit can be presented as Formula (10).

$$P^k = P_2^k + R^k =$$
$$= C_b(S(C_a(l_{k-1}), 1-\tfrac{1}{s})) + C(S(C_a(l_{k-1}), \tfrac{1}{s}), l_{k-1}) \quad (10)$$

Wherein: P_k is the output of the enhancement unit.

Normalized using batch normalization

Batch normalization is applied to each individual layer (or potentially to all layers) and operates as follows: during each training iteration, at each layer, the activation values are first computed as usual. Then, the activation values for each node are normalized by subtracting the mean and dividing by the standard deviation. Both of these quantities are estimated based on the statistics of the current minibatch [34].

The output feature map is defined as Formula (11).

$$Y_{ijkt} = W_k \frac{X_{ijkt} - \mu_k}{\sqrt{\sigma_k^2 + \epsilon}} + b_k$$
$$\mu_k = \frac{1}{HWT} \sum_{i=1}^{H} \sum_{j=1}^{W} \sum_{t=1}^{T} X_{ijkt} \quad (11)$$
$$\sigma_k^2 = \frac{1}{HWT} \sum_{i=1}^{H} \sum_{j=1}^{W} \sum_{t=1}^{T} (X_{ijkt} - \mu_k)^2$$

Wherein, the input and output arrays are considered as 4D tensors to work with a series of feature maps. The tensors w and b define the multiplicative and additive constants, respectively. T is the batch size for batch normalization and:

$$X, Y \in \mathcal{R}^{H \times W \times K \times T}, W \in \mathcal{R}^{K}, b \in \mathcal{R}^{K}$$

III. EXPERIMENTATION AND EVALUATION OF RESULTS

Experimentation

The implementation of image super-resolution, denoising, and restoration was carried out using a proposed neural network on the collected dataset, which include images of cars, motorcycles, and street scenes with three RGB color channels, each sized 3 x 256 x 256. The dataset included 685 training images and 170 test images (Figure 9).

Fig. 9: A few images in the dataset collected by Authors

The process involved the following steps:

Step 1 – Data preparation: Generating low-resolution images from high-resolution references by adding noise and applying a random erasure function to simulate content loss in the images. Step 2 – Initial image enhancement: Applying super-resolution to the low-resolution images and perform denoising and content restoration using bilinear interpolation as the preliminary enhancement method.

Step 3 – Advanced image restoration: Implementing the proposed neural network to perform super-resolution, denoising, and restoration tasks on the input images, which is anticipated to improve the image quality significantly.

Step 4 – Visual evaluation: Conducting a visual assessment of the high-resolution output produced by the neural network model. This step involves a comparative analysis of the images restored with pretrained weights against those enhanced by traditional methods.

Step 5 – Quality assessment: Evaluating the restored images quantitatively by computing similarity metrics with the original high-resolution images. Key metrics include Train PSNR, Train loss, Valid PSNR, Valid loss, and Test PSNR.

Step 6 – Performance comparison: Finally, benchmark the execution time of the neural network against other methods to evaluate the efficiency and practicality of the proposed solution in real-world scenarios.

During the training phase, the proposed network receives input images of dimensions $3 \times 256 \times 256$. The iterative optimization algorithm, gradient descent, is employed with a learning rate set to 0.001. The training process spans across a total of 100 epochs.

Visual evaluation

The results of the visual evaluation using the proposed method are presented in Figure 10. The image displays results from experiments

using different configurations of the ED-Unet model for image processing, specifically focusing on enhancing image resolution or restoration. Configurations utilizing skip connections and pretrained weights showed significant improvements in image clarity and quality compared to the model without skip connections. The performance graphs, likely measuring PSNR and SSIM, indicate that adding skip connections and pretrained weights significantly enhances the model's ability to produce high-quality images from lower-resolution inputs.

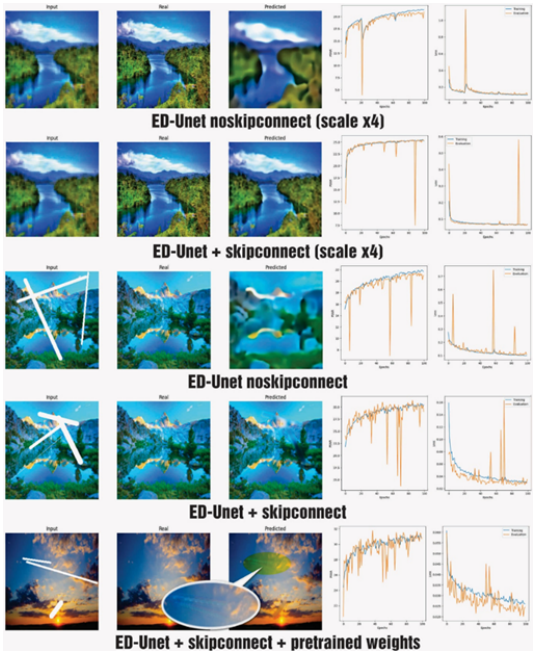

Fig. 10: Visualization through images and data visualization

It can be observed that the outcomes based on the proposed neural network model and the method using pretrained weights exhibit the highest image quality. The restored images with the proposed method nearly fully recover all features and content, with enhanced visibility of details that were previously noisy or lacked information. Furthermore, data visualization using the PSNR index shows that higher PSNR values indicate better image enhancement results.

Evaluating image quality metrics

When analyzing the performance of the ED-Unet model in various configurations as listed in Table 3, the use of pretrained weights particularly stands out due to the significant benefits it offers. Specifically, the 'ED-Unet + skip connect + pretrained weights' configuration demonstrates superior improvement over other setups, both in terms of performance and efficiency.

In the context of practical applications, where high-resolution image enhancement requires precision and quick response times, using pretrained weights not only improves image quality but also ensures that the model can be deployed effectively. The combination of skip connection and pretrained weights creates an exceptional model capable of handling the challenges of image resolution in situations requiring complex computations. This not only showcases the model's capabilities in a controlled experimental environment but also in practical applications, where detail recovery and noise reduction are important.

Table 3: Image quality assessment metrics by method (epoch = 100)

Model name	Scale	Train PSNR	Train loss	Valid PSNR	Valid loss	Test PSNR
ED-Unet noskipconnect	x4	21.427	0.104	20.954	0.112	20.9536
ED-Unet + skipconnect	x4	25.188	0.067	25.321	0.065	25.4323
ED-Unet noskipconnect		21.671	0.100	21.047	0.109	21.3795
ED-Unet + skipconnect		30.459	0.030	29.031	0.035	30.6391
ED-Unet + skipconnect + pretrained weight		30.791	0.026	30.699	0.025	31.0172

IV. CONCLUSION

Recent research and testing have demonstrated that the proposed network significantly excels in denoising and enhancing the resolution of real-world images. This network, with its deep and elaborate structure, can understand and process various noise types directly from input images without prior noise information. This capability is attributed to the network's numerous convolutional layers, which enable it to detect and decode complex noises like Gaussian or Poisson.

Simpler networks with fewer layers would not be able to efficiently accomplish this task due to a lack of necessary parameters to handle noise complexity. With its wide receptive field, our network not only removes noise but also accurately reconstructs image details.

Moreover, the training and execution time of this solution is reasonably short, emphasizing its potential for broad application in the field of image processing. This opens up opportunities for our network to be used not only in experimental settings but also in real-world applications where image quality enhancement is critical.

REFERENCES

[1] Janani P. Image quality enhancement techniques for digital media. *Indian Journal of Science and Technology*. 2015;8(22): 1–12. https://doi.org/10.17485/ijst/2015/v8i22/79318.

[2] He K, Zhang X, Ren S, Sun S. Deep residual learning for image recognition. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 27^{th}–30^{th} June 2016; Las Vegas, NV, USA. IEEE; 2016. p.770–778. https://doi.org/10.1109/CVPR.2016.90.

[3] Wang Z, Chen J, Hoi S. Deep learning for image super-resolution: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. 2021;43(10): 3365–3387. https://doi.org/10.1109/TPAMI.2020.2982166.

[4] Liang X, Chen X, Ren K. Low-light image enhancement via adaptive frequency decomposition network. *Scientific Reports*. 2023;13(1): 14107. https://doi.org/10.1038/s41598-023-40899-8.

[5] Hüsem H, Orman Z. A survey on image super-resolution with generative adversarial networks. *Acta Infologica*. 2020:4(2): 139–154. https://doi.org/10.26650/acin.765320.

[6] Moghimi MK, Mohanna F. Real-time underwater image resolution enhancement using super-resolution with deep convolutional neural networks. *Journal of Real-Time Image Processing*. 2021;18(5): 1653–1667. https://doi.org/10.1007/s11554-020-01024-4.

[7] Dong C, Loy CC, He K, Tang X. Learning a deep convolutional network for image super-resolution. In: Fleet D, Pajdla T, Schiele B, Tuytelaars T (eds.). *Computer Vision – ECCV 2014*. Cham: Springer; 2014. https://doi.org/10.1007/978-3-319-10593-2_13.

[8] Kim J, Lee JK, Lee KM. Deeply-recursive convolutional network for image super-resolution. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 27^{th}–30^{th} June 2016; Las Vegas, NV, USA. IEEE; 2016. p.1637–1645. https://doi.org/10.1109/CVPR.2016.181.

[9] Dong C, Loy CC, Tang X. Accelerating the super-resolution convolutional neural network. In: Leibe B, Matas J, Sebe N, Welling M (eds.). *Computer Vision – ECCV 2016*. Cham: Springer; 2016. https://doi.org/10.1007/978-3-319-46475-6_25.

[10] Shi W, Caballero J, Huszár F, Totz J, Aitken AP, Bishop R, et al. Real-time single image and video super-resolution using an efficient subpixel convolutional neural network. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 27^{th}–30^{th} June 2016; Las Vegas, NV, USA. IEEE; 2016. p.1874–1883. https://doi.org/10.1109/CVPR.2016.207.

[11] Ledig C, Theis L, Huszár F, Caballero J, Cunningham A, Acosta A, et al. Photo-realistic single image super-resolution using a generative adversarial network. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 21^{st}–26^{th} July 2017; Honolulu, HI, USA,. IEEE; 2017. p.105–114. https://doi.org/10.1109/CVPR.2017.19.

[12] Lim B, Son S, Kim H, Nah S, Lee KM. Enhanced deep residual networks for single image super-resolution. In: *IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*. 21^{st}–26^{th} July 2017; Honolulu, HI, USA. IEEE; 2017. p.1132–1140. https://doi.org /10.1109/CVPRW.2017.151.

[13] Sajjadi MS, Scholkopf B, Hirsch M. EnhanceNet: Single image super-resolution through automated texture synthesis. In: *IEEE International Conference on Computer Vision (ICCV)*. 22^{nd}–29^{th} October 2017; Venice, Italy. IEEE; 2017. p.4501–4510. https://doi.org/10.1109/ICCV .2017.481.

[14] Tai Y, Yang J, Liu X, Xu C. MemNet: A persistent memory network for image restoration. In: *IEEE International Conference on Computer Vision (ICCV)*. 22^{nd}–29^{th} October 2017; Venice, Italy. IEEE; 2017. p.4549–4557. https://doi.org/10.1109/ICCV.2017.486.

[15] Tong T, Li G, Liu X, Gao Q. Image super-resolution using dense skip connections. *IEEE International Conference on Computer Vision (ICCV)*. 22^{nd}–29^{th} October 2017; Venice, Italy. IEEE; 2017. p.4809–4817. https://doi.org/10.1109/ICCV.2017.514.

[16] Zhang Y, Tian Y, Kong Y, Zhong B, Fu Y. Residual dense network for image super-resolution. In: *IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 18^{th}–23^{rd} June 2018; Salt Lake City, UT, USA. IEEE; 2018. p.2472–2481. https://doi.org/10.1109/CVPR.2018.00262.

[17] Ahn N, Kang B, Sohn KA. Fast, accurate, and lightweight super-resolution with cascading residual network. In: Ferrari V, Hebert M, Sminchisescu C, Weiss Y (eds.). *Computer Vision – ECCV 2018*. Cham: Springer; 2018. https://doi.org/10.1007/978-3-030-01249-6_16.

[18] Zhang Y, Li K, Li K, Wang L, Zhong B, Fu

Y. Image super-resolution using very deep residual channel attention networks. In: Ferrari V, Hebert M, Sminchisescu C, Weiss Y (eds.) *Computer Vision – ECCV 2018*. Cham: Springer; 2018. https://doi.org/10.1007/978-3-030-01234-2_18.

[19] Wang X, Yu K, Wu S, Gu J, Liu Y, Dong C, et al. ESRGAN: Enhanced super-resolution generative adversarial networks. In: Leal-Taixé L, Roth S (eds.). *Computer Vision – ECCV 2018 Workshops*. Cham: Springer; 2018. https://doi.org/10.1007/978-3-030-11021-5_5.

[20] Zhang Y, Li K, Li K, Zhong B, Fu Y. Residual non-local attention networks for image restoration. *arXiv* [Preprint] 2019. https://doi.org/10.48550/arXiv.1903.10082.

[21] Dai T, Cai J, Zhang Y, Xia ST, Zhang L. Second-order attention network for single image super-resolution. In: *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 15^{th}–20^{th} June 2019; Long Beach, CA, USA. IEEE; 2019. p.11057–11066. https://doi.org/10.1109/CVPR .2019.01132.

[22] Niu B, Wen W, Ren W, Zhang X, Yang L, Wang S, et al. Single image super-resolution via a ternary attention network. *Applied Intelligence*. 2023;53(11): 13067–13081. https://doi.org /10.1007/s10489-022-04129-4.

[23] Chen H, Wang Y, Guo T, Xu C, Deng Y, Liu Z, et al. Pre-trained image processing transformer. In: *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 20^{th}–25^{th} June 2021; Nashville, TN, USA. IEEE; 2021. p.12294–12305. https://doi.org/10.1109/CVPR46437 .2021.01212.

[24] Liang J, Cao J, Sun G, Zhang K, Van Gool L, Timofte R. SwinIR: Image restoration using swin transformer. In: *IEEE/CVF International Conference on Computer Vision Workshops (ICCVW)*. 11^{th}–17^{th} October 2021; Montreal, BC, Canada. IEEE; 2021. p.1833–1844. https://doi.org/10.1109 /ICCVW54120.2021.00210.

[25] Xia B, Zhang Y, Wang S, Wang Y, Wu X, Tian Y, et al. DiffIR: efficient diffusion model for image restoration. In: *IEEE/CVF International Conference on Computer Vision (ICCV)*. 01^{st}–06^{th} October 2023; Paris, France. IEEE; 2023. p.13049–13059. https://doi.org/10.1109/ICCV 51070.2023.01204.

[26] Zhang K, Zuo W, Chen Y, Meng D, Zhang L. Beyond a Gaussian Denoiser: Residual learning of deep CNN for image denoising. *IEEE Transactions on Image Processing*. 2017;26(7): 3142–3155. https://doi.org/10.1109/TIP.2017.2662206.

[27] Wang Z, Jian C, Steve CHH. Deep learning for image super-resolution: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. 2021;43(10): 3365–3387. https://doi.org/10.1109/TPAMI.2020.2982166.

[28] Bashir SMA, Wang Y, Khan M, Niu Y. A comprehensive review of deep learning-based single image super-resolution. *PeerJ Computer Science*. 2021;7: e621. https://doi.org/10.7717/peerj-cs.621.

[29] Odena A, Dumoulin V, Olah C. Deconvolution and checkerboard artifacts. *Distill*. 2016;1(10): e3. http://doi.org/10.23915/distill.00003.

[30] Krishnamoorthy DN, Vigneshraaja M. Enhancing the image super-resolution via U-net architecture for improved visual quality. *Turkish Journal of Physiotherapy and Rehabilitation*. 2021;32(2): 1073–1080.

[31] Parkhi OM, Vedaldi A, Zisserman A, Jawahar CV. *The Oxford-IIIT pet dataset*. https://www.robots.ox.ac.uk/ vgg/data/pets/ [Accessed 3^{rd} April 2024].

[32] Zhao H, Gallo O, Frosio I, Kautz J. Loss functions for image restoration with neural networks. *IEEE Transactions on Computational Imaging*. 2017;3(1): 47–57. https://doi.org/10.1109/TCI .2016.2644865.

[33] Zha L, Yang Y, Lai Z, Zhang Z, Wen J. A lightweight dense connected approach with attention on single image super-resolution. *Electronics*. 2021;10(11): 1234. https://doi.org/10.3390/electronics10111234.

[34] Ioffe S, Szegedy C. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In: *Proceedings of the 32^{nd} International Conference on Machine Learning*. 6^{th}–11^{th} July 2015; Lille, France. JMLR: W&CP; 2015. p.448–456.