

# UNet—: Memory-Efficient and Feature-Enhanced Network Architecture based on U-Net with Reduced Skip-Connections

Lingxiao Yin<sup>\*1</sup>, Wei Tao<sup>1</sup>, Dongyue Zhao<sup>1</sup>, Tadayuki Ito<sup>2</sup>, Kinya Osa<sup>2</sup>,  
 Masami Kato<sup>2</sup>, and Tse-Wei Chen<sup>2</sup>

<sup>1</sup> Canon Innovative Solution (Beijing) Co., Ltd.

{Lingxiao\_Yin, taowei, zhaodongyue}@canon-is.com.cn

<sup>2</sup> Device Technology Development Headquarters, Canon Inc.  
 twchen@ieee.org

**Abstract.** U-Net models with encoder, decoder, and skip-connections components have demonstrated effectiveness in a variety of vision tasks. The skip-connections transmit fine-grained information from the encoder to the decoder. It is necessary to maintain the feature maps used by the skip-connections in memory before the decoding stage. Therefore, they are not friendly to devices with limited resource. In this paper, we propose a universal method and architecture to reduce the memory consumption and meanwhile generate enhanced feature maps to improve network performance. To this end, we design a simple but effective Multi-Scale Information Aggregation Module (MSIAM) in the encoder and an Information Enhancement Module (IEM) in the decoder. The MSIAM aggregates multi-scale feature maps into single-scale with less memory. After that, the aggregated feature maps can be expanded and enhanced to multi-scale feature maps by the IEM. By applying the proposed method on NAFNet, a SOTA model in the field of image restoration, we design a memory-efficient and feature-enhanced network architecture, UNet—. The memory demand by the skip-connections in the UNet— is reduced by 93.3%, while the performance is improved compared to NAFNet. Furthermore, we show that our proposed method can be generalized to multiple visual tasks, with consistent improvements in both memory consumption and network accuracy compared to the existing efficient architectures.

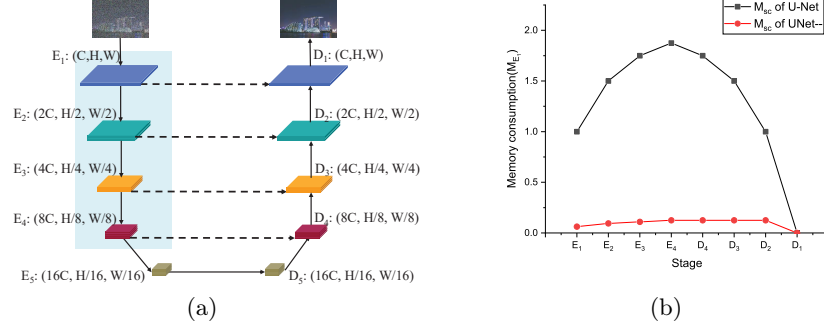
**Keywords:** Skip-connection · Memory · U-Net.

## 1 Introduction

U-Net [23] and its variants [12, 22, 28, 43, 44] have become the de facto standard in the field of pixel-level computer vision research [41]. With the ability to learn effective features efficiently and robustly, they achieve superior performance in

---

<sup>\*</sup> Corresponding author.



**Fig. 1:** (a) Architecture of a common U-Net model. The dashed lines represent the skip-connections. Feature maps in the blue shadow are needed to be maintained in memory. (b) Memory consumption for skip-connections for U-Net and UNet— throughout the inference process, the unit is  $M_{E_1}$  which denotes the memory consumption of  $E_1$ .

many tasks *e.g.*, image restoration [5], image segmentation [4, 6, 10], image matting [36], *etc.* U-Net consists of an encoder, a decoder, and skip-connections between them as in Fig. 1(a). Feature maps are downsampled stage by stage in the encoding phase and correspondingly upsampled in the decoding phase. The skip-connections transmit the details of the encoding information to the decoder. They can compensate for details loss due to downsampling operations with a tolerable computation cost. However, they are not friendly in terms of memory consumption in especially resource-limited device. The feature maps transmitted by the skip-connections require on-chip memory and the memory will not be released until the feature maps of the skip-connections are fused into the decoder. Therefore, despite their great success, deploying U-Net or its variants on resource-limited devices is still a non-trivial task.

To reduce the heavy memory consumption induced by the skip-connections, recently many researchers have proposed methods of efficient model design to attempt to merge them into the parallel convolutional structures. RepVGG [7] proposes structural re-parameterization to decouple a training-time multi-branch topology with an inference-time plain architecture. Inspired by the method, the authors in [8, 29, 30] merge the skip-connections into a parallel convolutional structure during inference. The final obtained single-branch structure is equivalent to the original multi-branch structure, as there are no non-linear units in the original structure. Although they reduce the memory consumption and improve inference speed, the methods are not appropriate for the skip-connections in U-Net yet. The main reason is that in general in the multi-branch structure represented by the skip-connections, one of the branches on the main trunk contains several non-linear transformations. Consequently, it is not achievable to merge the skip-connections equally.

The authors in [25,32] propose to reconstruct a model without skip-connections which is obtained by progressively removing skip-connections during finetuning. The model is trained with the distillation method to mitigate performance drop, where the teacher model is combined with skip-connections. Here the skip-connections often refer to the shortcuts in residual structure, where the distance between layers connected by skip connections is approximately two or three layers. In contrast, the layers which are connected by the skip-connections in U-Net are very far away. It spans across two stages of the encoder and the decoder, and it often leads to a large difference in the feature maps at both endpoints of the skip-connections. Directly removing such skip-connections from U-Net can result in significant accuracy loss, which is verified in Tabs. 4 to 7. It is important to reduce the memory consumption of the skip-connections without introducing harm in accuracy.

Some other works aim to improve U-Net performance by designing complex structures [14, 35, 38, 39] to replace the simple skip-connections. The authors in [14] found that the combination of feature maps from multiple encoder levels generates enriched features compared to the simple skip-connection. This benefits from the combination of various regions of interest from different levels. However, these complex structures lead to high computation and memory usage. We propose an efficient multi-scale information aggregation module (MSIAM) and an effective information enhancement module (IEM) which can be easily integrated into any U-Net or its variants. Specifically, the MSIAM first reduces the number of channels in multi-scale feature maps to save memory, then these compressed feature maps are resized to a single-scale with the same resolution and concatenated together, and finally information interaction is achieved through point-wise convolution. The generated single-scale feature map is maintained in memory instead of the original large feature maps. The IEM module is designed to generate enhanced multi-scale feature maps based on the single-scale feature map. First, the single-scale feature map is resized to multiple groups of feature maps with different resolutions, and then an enhancement block is applied to each group to generate enriched feature maps. To avoid a large computational cost, we introduce a ConvNeXt V2 block and a separable convolution in the enhancement block.

By applying MSIAM and IEM to U-Net, we construct a memory-efficient and feature-enhanced network architecture named UNet—.

The main contributions are summarized as follows:

1. We significantly reduce the memory consumption induced by the skip-connections in U-Net and its variants, and further construct a memory-efficient and feature-enhanced network architecture.
2. We design plug-and-play modules, MSIAM and IEM, which are applicable to any U-Net and its variants.
3. We demonstrate the proposed method in a great variety of tasks and verify its generality and effectiveness.

## 2 Related work

The skip-connection, also known as shortcut or residual, is a popular trick in neural network design. It was put forward by ResNet [11] in which the layer is reformulated as residual learning function with reference to the input of the layer by introducing an identity mapping. The design of skip-connections makes training deeper networks easier and gains accuracy from the increased depth. Inspired by this idea, the skip-connection is widely used in latter networks [27]. Inception-v4 [27] significantly accelerates Inception network training by skip-connections. ResNeXt [34] combines skip-connections with designed homogeneous multi-branch architecture. Res2Net [9] constructs hierarchical residual-like connections within one single residual block.

Based on FCN [24], U-Net [23] introduces long-path skip-connections between the contracting path and the expansive path. Such network design obtains great success in pixel-level prediction task (e.g. semantic segmentation, instance segmentation, image restoration) as the skip-connections can transmit fine-grained information lost caused by downsampling. In order to obtain higher accuracy, some designs build more complex connections. [12, 43], introduce attention mechanism for the feature maps of skip-connections [15, 22, 26]. Some recent work designs hybrid block to model global and multi-scale context [31], [14].

Although we can benefit from skip-connections, large memory consumption is required. These works [7, 29, 30] demonstrate that through structural re-parameterization, skip-connections can be merged into their parallel convolutions to reduce extra memory requirements. [7] is a simple architecture with a stack of  $3 \times 3$  conv and ReLU. During the training process, it stacks a sets of multi-branch structures composed of  $3 \times 3$  conv,  $1 \times 1$  conv, and identity mapping. After training, it converts the trained multi-branch structure into a single  $3 \times 3$  conv for inference. Consequently, the converted model only has a stack of  $3 \times 3$  conv, which is efficient for test and deployment. [30] is another technique that converts multi-branch networks with skip-connections to plain networks through structural re-parameterization and over-parameterization. [29] uses structural re-parameterization to lower the memory access cost by removing skip-connections in the network based on transformer architecture. Some other works are proposed to progressively remove skip-connections during training process [32] or further with distillation [25]. [32] introduces two new methods, SkipRemover and SkipShortener, that alters networks with skip-connections dynamically during retraining by remove or shorten the skip connections to fit better on hardware, achieving resource-efficient inference with minimal to no loss in accuracy. As analyzed in Sec. 1, these methods are not applicable to skip-connections with long path in U-Net networks. In this paper, we propose a different method to significantly decrease memory consumption caused by skip-connections, a set of multi-scale feature maps are transformed into a lightweight single-scale one, which will be expanded and enhanced later.



### 3 Methodology

In this chapter, we introduce the runtime memory analysis of common U-Net models in Sec. 3.1. We present the proposed UNet— architecture and its details in Sec. 3.2 and Sec. 3.3.

#### 3.1 Memory analysis

The common U-Net model and its runtime memory analysis is shown in Fig. 1. There are five stages in the encoder and decoder, respectively. The feature maps output at each stage in the encoding process are marked as  $\{E_1, E_2, E_3, E_4, E_5\}$ . Between adjacent stages, by downsampling operations the feature map size is halved and the number of channels is doubled. Similarly, the feature maps output at each stage in the decoding process are marked as  $\{D_1, D_2, D_3, D_4, D_5\}$ . Between adjacent stages, through upsampling operations, the feature map size is doubled and the number of channels is halved. There are four skip-connections that transmit the feature maps from  $E_1$  to  $D_1$ ,  $E_2$  to  $D_2$ ,  $E_3$  to  $D_3$ , and  $E_4$  to  $D_4$ , respectively. We note the total memory size for feature maps to be transmitted via skip-connection as  $M_{sc}$ . To simplify the analysis, we ignore the memory required for loading model parameters as its value is constant. We consider the value of  $M_{sc}$  at the end of each stage of the encoder and decoder processes. At the end of  $E_1$ , the output feature maps need to be maintained in memory for later connection to  $D_1$ . The size of the feature maps is calculated as Eq. (1).

$$M_{E_1} = C \times H \times W \quad (1)$$

then  $M_{sc} = M_{E_1}$  at the end of  $E_1$ . Similarly at the end of  $E_2$ , the output feature maps also need to be maintained for later connection to  $D_2$  whose memory size is:

$$M_{E_2} = 2 \times C \times \frac{H}{2} \times \frac{W}{2} = \frac{M_{E_1}}{2} \quad (2)$$

then  $M_{sc} = M_{E_1} + M_{E_2} = \frac{3 \times M_{E_1}}{2}$ . Analogously,  $M_{sc} = M_{E_1} + M_{E_2} + M_{E_3} = \frac{7 \times M_{E_1}}{4}$  at the end of  $E_3$ , and  $M_{sc} = M_{E_1} + M_{E_2} + M_{E_3} + M_{E_4} = \frac{15 \times M_{E_1}}{8}$  at the end of  $E_4$ . The memory of maintaining feature maps keeps increasing in the encoder process. It reaches a peak value of  $\frac{15}{8} \times M_{E_1}$  at the end of  $E_4$ . At the end of  $D_4$ , the maintained feature maps from  $E_4$  is released. Thus, at that time  $M_{sc}$  starts to be decreased to  $M_{E_1} + M_{E_2} + M_{E_3}$ . Along with the entire decoding process,  $M_{sc}$  continues to be decreased to zero until the end the  $D_1$ . We illustrate the temporal distributive characteristics of  $M_{sc}$  in Fig. 1(b). It is worth to note that the  $M_{E_4}$  is only  $M_{E_1} \times \frac{1}{8}$ . That means the skip-connections represented by  $M_{sc}$  consume extra 14 times memory of  $M_{E_4}$  when it reaches peak value. Most of the memory will not be released until the decoding process is finished. In conclusion, the skip-connections present a great challenge for model deployment in resource-limited devices.

Models	PSNR	SSIM
Model-0 (w/o skip-connection)	39.4737	0.9561
Model-1 ( $E_1 \rightarrow D_1$ )	39.7906	0.9587
Model-2 ( $E_2 \rightarrow D_2$ )	39.8738	0.9593
Model-3 ( $E_3 \rightarrow D_3$ )	39.8703	0.9593
Model-4 ( $E_4 \rightarrow D_4$ )	39.8111	0.9587

**Table 1:** Performance comparison with only one skip-connection, where  $(E_i \rightarrow D_i)$  represents the skip-connection connects  $E_i$  and  $D_i$ . The results are evaluated on the SIDD validation dataset.

### 3.2 Overview of the method

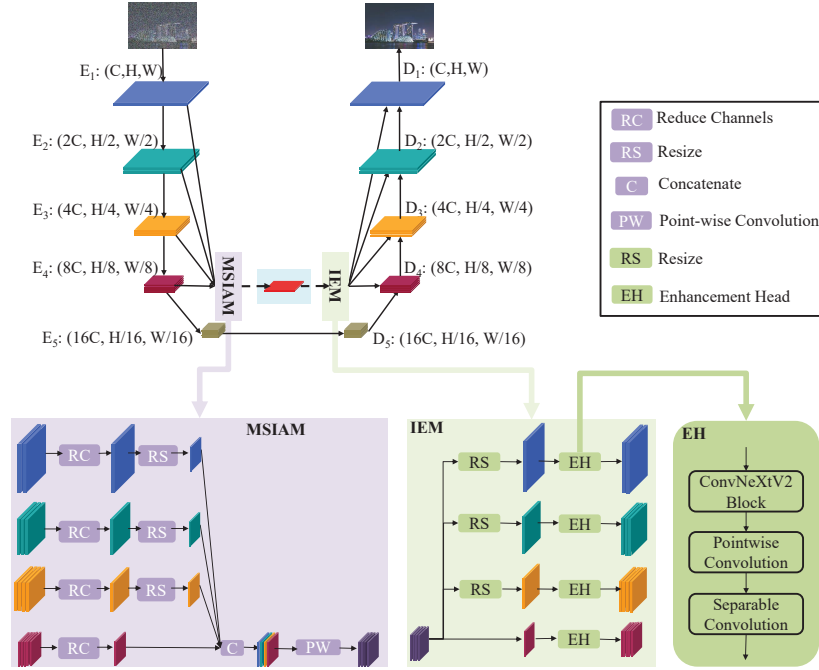
We propose a universal method to reduce memory consumption caused by the skip-connections and meanwhile generate enhanced feature maps which boost network performance. Applied with our method, we design a memory-efficient and feature-enhanced network architecture based on U-Net, UNet—, which is named for consuming less memory by skip-connections than U-Net. The model architecture is shown in Fig. 2.

There are two main modules in UNet—: Multi-Scale Information Aggregation Module (MSIAM) and Information Enhancement Module (IEM). Unlike U-Net models, in UNet— the multi-scale feature maps output from the encoder are not directly transmitted to the decoder. They are fed into the proposed MSIAM and become lighter sequentially. MSIAM aggregates the light-weight multi-scale feature maps and generates the corresponding single-scale feature maps but with information from multiple receptive fields. The single-scale feature maps require less memory than common skip-connections. During the decoding process, according to the input single-scale feature maps, IEM can generate a group of multi-scale feature maps. The generated multi-scale feature maps carry enhanced information with stronger representative ability. Through this way, the memory demand for skip-connections is significantly reduced to single-scale feature maps and the enhanced information generated by IEM helps to improve the model performance. The macro architecture of the proposed modules is described in the following section.

### 3.3 Macro architecture design

**Multi-Scale Information Aggregation Module (MSIAM)** The target of this module is to generate a group of single-scale, compact, and representative feature maps. The memory footprint of the feature maps should be much smaller than that of common skip-connections.

To reduce the memory consumption caused by skip-connections, we are considering whether each skip-connections is crucial and whether it is possible to remove certain skip-connections directly. We first study the contribution of each skip-connections. We construct four U-Net models; each model has only one



**Fig. 2:** Illustration of the proposed UNet— network. The original skip-connections are replaced with MSIAM and IEM. MSIAM aggregates multi-scale feature maps to single-scale with less memory demand. IEM generates enhanced multi-scale feature maps according to the output of MSIAM. Feature maps in the blue shadow are needed to be maintained in memory.

skip-connection between encoder and decoder. The four U-Net models share the same architecture except the position of the single skip-connection. Specifically, the single skip-connection in Model-1 connects  $E_1$  and  $D_1$ , the skip-connection in Model-2 connects  $E_2$  and  $D_2$ , the skip-connection in Model-3 connects  $E_3$  and  $D_3$ , and the skip-connection in Model-4 connects  $E_4$  and  $D_4$ . Besides, we also construct Model-0 without any skip-connections for comparison. We verify the performance of these five models in the image denoising task. The results of the experiment are shown in Tab. 1. We find that the contribution ratio is different and irregular. This is similar to the findings in [31] where they found that the optimal combination of skip-connections is different for different datasets. To obtain a universal structure which is appropriate with multiple tasks and datasets, it is useless to find an optimal combination of skip-connections. In contrast, we should keep all skip-connections to avoid the performance dropping dramatically.

In addition, the skip-connections in U-Net are independent of each other, which leads to the information with different receptive fields lacking interactivity. Referring to the ideas in [12, 14, 17, 31, 43], we think that it is potential to generate

powerful and robust information by combining all these multi-scale feature maps and directing the information to cooperate with each other automatically.

Taking these considerations into account, we design MSIAM in Fig. 2. There are four units in MSIAM that can be represented as Eq. (3).

$$E' = \text{PWConv}(\text{RS}(\text{RC}(E_1)) || \text{RS}(\text{RC}(E_2)) || \dots || \text{RS}(\text{RC}(E_n)) || \dots || \text{RS}(\text{RC}(E_N))) \quad (3)$$

where,  $E'$  is single-scale output feature maps of MSIAM,  $N$  and  $n$  represents how many stages in encoder and the current stage index, respectively,  $E_n$  represents the feature maps outputted by the  $n$ th stage,  $\text{PWConv}$  is a point-wise convolution,  $\text{RS}$  means the resizing operation which adjusts the resolution of feature maps,  $\text{RC}$  means reducing channels of feature maps,  $||$  represents the concatenation operation.

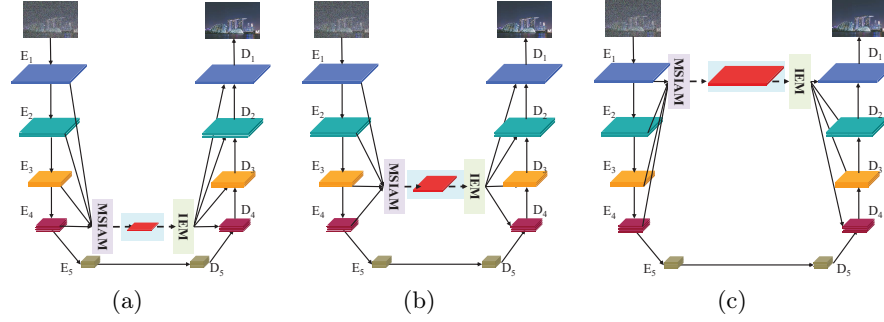
Firstly, the multi-scale feature maps is pruned to be slim. It can be realized by point-wise convolution to avoid a large computational cost. Then these multi-scale feature maps need to be resized to have the same resolution. The target resolution can be any of the multi-scale feature maps generated by the end of the encoding process. We illustrate three classical types of architecture that correspond to different target resolutions in Fig. 3. For model in Fig. 3(a), the resizing is performed by Pixel Unshuffle, for model in Fig. 3(c), the resizing is realized by Pixel Shuffle, and for model in Fig. 3(b), the resizing is realized by both. The resized feature maps are further concatenated along the channel dimension. Finally, the information interaction is performed by a point-wise convolution  $\text{PWconv}$ .

The parameters in the MSIAM are optimized during the training process, which are used to weight different feature maps. In this way, the information ratio between different feature maps will automatically be adjusted. Therefore, multi-scale fine-grained details are interactive and complementary to each other, leading to compact but enriched feature maps.

**Information Enhancement Module (IEM)** IEM is designed to generate enhanced multi-scale feature maps according to the compact single-scale feature maps produced by MSIAM.

As shown in Fig. 2, IEM consists of a resize operation and an enhancement header. Firstly, the input single-scale feature map is resized to a specific resolution according to the current decoder stage. The resize operation can be an unsampling or a downsampling operator, which depends on the relative sizes of the resolutions between source and target feature maps. Accordingly, there are three classical types of IEM in Fig. 3. For the network architecture in Fig. 2, the resize operation is applied with the pixel shuffle operation. The enhancement header consists of a ConvNeXt V2 [33] block and a separable convolution. We apply the ConvNeXt V2 block due to its powerful representative ability. The separable convolution adjusts the number of output channels, which should be consistent with the channels output in the corresponding encoding stage.

To verify that the generated feature maps by IEM are not imitations of the feature maps output by the encoding process but with enhanced information,



**Fig. 3:** Model structure variants corresponding to different target resolution. Feature maps in the blue shadow are needed to be maintained in memory. (a) The resolution of the aggregated feature maps equals to the minimum resolution in the encoder. (b) The resolution of the aggregated feature maps equals to the intermediate resolution in the encoder. (c) The resolution of the aggregated feature map equals to the maximum resolution in the encoder.

**Table 2:** The similarity relationship between the output feature maps of encoder and IEM. A larger SSIM represents greater similarity.

Stage	SSIM
$\{E_1, IEM_1\}$	0.1813
$\{E_2, IEM_2\}$	0.0179
$\{E_3, IEM_3\}$	0.0131
$\{E_4, IEM_4\}$	0.0157

**Table 3:** Comparison of representative ability between the output feature maps of encoder and IEM, where RA stands for the representative ability.

Stage	RA of Encoder	RA of IEM
1	0.3315	0.4185
2	0.2768	0.6529
3	0.2203	0.3662
4	0.2209	0.2460

we first compute the similarity between these two groups of feature maps. We apply SSIM [42] as the similarity metric, within the range of [0,1]. The result is shown in Tab. 2. It shows that the feature maps generated by IEM are different from the encoder outputs.

To compare the representative ability of feature maps, the method used in [33] is modified. For the feature maps generated by the encoder and IEM, we calculate the cosine distance between any two channels and then calculate the mean and variance values of these distances, respectively. Based on similar mean values, the larger the variance, the greater the difference between channels, indicating a richer diversity in feature maps. The result is shown in Tab. 3. We can observe that the feature maps generated by IEM is embedded with enhanced information which will boost the model’s performance.

## 4 Experiments and Analysis

**Details of NAFNet with UNet—** We apply the proposed UNet— into NAFNet [5], which is one of the SOTA models in the field of image restoration. NAFNet adopts a typical U-Net structure, with the encoder consisting of 5 stages  $\{E_1, E_2, E_3, E_4, E_5\}$ . Among them, the feature maps outputted by the first 4 stages need to be connected to the decoder via skip connections. They respectively output feature maps with channel numbers of  $\{32, 64, 128, 256\}$  and resolutions of  $\{H * W, \frac{H}{2} * \frac{W}{2}, \frac{H}{4} * \frac{W}{4}, \frac{H}{8} * \frac{W}{8}\}$ . Here,  $H$  and  $W$  represent the height and width of the input images.

In MSIAM, using the reduction ratio of  $\{\frac{1}{16}, \frac{1}{16}, \frac{1}{16}, \frac{1}{8}\}$ , we first decrease the number of channels output by the first 4 stages to  $\{2, 4, 8, 32\}$ , respectively. Then we apply pixel unshuffle operators with stride =  $\{8, 4, 2\}$  to decrease resolutions of  $\{E_1, E_2, E_3\}$  to  $\frac{H}{8} * \frac{W}{8}$ . Finally, to obtain single-scale feature maps we concatenate them with  $E_4$  along the channel dimension, and further aggregate all data through point-wise convolution.

In IEM, according to the single-scale input feature maps generated by MSIAM, we first use pixel shuffle operators with stride =  $\{8, 4, 2\}$  to restore the resolution of the first 3 stages  $\{E_1, E_2, E_3\}$ , respectively. Then, we use four enhancement header blocks to generate multi-scale feature maps separately on the input single-scale feature maps and the newly generated feature maps with higher resolution. The final generated multi-scale feature maps share the same channel number and resolution as  $\{E_1, E_2, E_3, E_4\}$ .

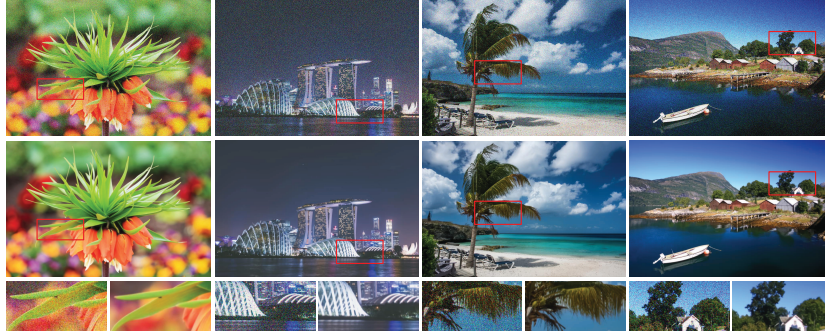
### 4.1 Image denoising

Noise can be introduced into an image during acquisition or processing. Image denoising aims to restore an image to its original quality by reducing or removing the noise, while preserving the important features of the image.

We use SIDD dataset [1] as our training and test dataset, which has 30,000 noisy images from 10 scenes under different lighting conditions using five representative smartphone cameras. These images are cropped non-overlapping and generated 31888 patches whose size is  $256 \times 256$ . These patches are divided into 30608 training patches and 1280 validation patches.

During the training process, the model is optimized with PSNR loss and Adam optimizer ( $\beta_1 = 0.9$ ,  $\beta_2 = 0.9$ , weight decay= 0) for total 200K iterations. The initial learning rate is  $1e^{-3}$  which is reduced to  $1e^{-7}$  with the cosine annealing schedule [18]. The training batch size is set to 32. We use the Peak-Signal-to-Noise Ratio (PSNR) and the Structural Similarity Index (SSIM) [42] as metrics to evaluate our models.

In Tab. 4, we first compare the proposed UNet— against the baseline model NAFNet and KBNNet, respectively. Our model reduces the memory consumption caused by skip-connections by 93.3%, however the performance is improved by 0.04 in PSNR in both experiments. It is worth noting that the MACs and the number of parameters increase only by 7.9%, 8%, 8.4% and 3.0%, respectively. This indicates that the proposed UNet— can reduce the memory



**Fig. 4:** Visualization results. The first row includes noisy images, the second row includes clean images outputted by NAFNet [5] with UNet—, the third row shows the comparison of local details.

cost of skip-connections without causing harm to the model performance in an efficient way. We also compare the proposed NAFNet with UNet— against the NAFNet without skip-connections. From Tab. 4, we can observe that our model outperforms the model without any skip-connection by 0.4 in PSNR and 0.003 in SSIM.

## 4.2 Image deblurring

Image deblurring is one of the image processing tasks that involves removing the blurring artifacts from images or videos to restore the original sharp content.

For image deblurring, we validate the performance of NAFNet with UNet— on GoPro [21] dataset. It provides pairs of realistic blurry images and the corresponding sharp ground-truth images that are obtained by a high-speed camera. The blurry image is produced by averaging successive latent frames, while the sharp image is defined as the mid-frame among the sharp frames that are used to make the blurry image. There are 3,214 blurred images with the size of  $1280 \times 720$  that are divided into 2,103 training images and 1,111 test images.

**Table 4:** Models and their performance in image denoising. SC is the abbreviation for skip connection. Output feature map of  $E_1$ :  $256 \times 256$ , 32 channels, 8bit.

Model	PSNR	SSIM	$M_{sc}$ (MB)	MACs(G)	Params(MB)
NAFNet [5]	39.97	0.960	3.75	16.23	29.16
NAFNet w/o SC	39.61	0.957	0.00	16.23	29.16
NAFNet with UNet—	40.01 ( $\uparrow 0.04$ )	0.960 ( $\uparrow 0.0$ )	0.25( $\downarrow 93.3\%$ )	17.52( $\uparrow 7.9\%$ )	29.98( $\uparrow 2.8\%$ )
KBNet [40]	40.35	0.962	7.5	58.19	104.93
KBNet with UNet—	40.39 ( $\uparrow 0.04$ )	0.962 ( $\uparrow 0.0$ )	0.5( $\downarrow 93.3\%$ )	63.11( $\uparrow 8.4\%$ )	108.12( $\uparrow 3.0\%$ )



**Table 5:** Models and their performance in image deblurring. SC is the abbreviation for skip connection. Output feature map of  $E_1$ :  $256 \times 256$ , 32 channels, 8bit.

Model	PSNR	SSIM	$M_{sc}$ (MB)	MACs(G)	Params(MB)
NAFNet [5]	32.87	0.960	3.75	16.23	29.16
Tailor [32]	32.87	0.960	0.00	16.23	29.16
NAFNet w/o SC	32.52	0.958	0.00	16.23	29.16
NAFNet with UNet—	33.06 ( $\uparrow 0.19$ )	0.962 ( $\uparrow 0.002$ )	0.25 ( $\downarrow 93.3\%$ )	17.52 ( $\uparrow 7.9\%$ )	29.98 ( $\uparrow 2.8\%$ )

**Fig. 5:** Visualization results. The first row includes blurry image, the second row includes clean images outputted by NAFNet [5] with UNet—, the third row shows the comparison of local details.

The evaluation result is shown in Tab. 5. With the same 93.3% memory savings, the proposed NAFNet with UNet— outperforms the baseline model with 0.19 in PSNR and 0.002 in SSIM.

### 4.3 Image super-resolution

Image super-resolution aims to increase the resolution of an image, while maintaining its content and details as much as possible. It can be used for various applications, such as improving image quality, enhancing visual detail, and increasing the accuracy of computer vision algorithms. We train the model using the exact same network as image denoising and image deblurring. We use DIV2K [2] and Flickr2K as the training dataset. We use five commonly used datasets including Set5 [3], Set14 [37], B100 [19], Urban100 [13] and Manga109 [20] as the test dataset. PSNR and SSIM are used to evaluate the quality of the restored images. We use the AdamW optimizer with  $\beta_1 = 0.9$  and  $\beta_2 = 0.99$  to solve the proposed model. The number of iterations is set to 500,000. We also set the initial learning rate to  $1e - 3$  and the minimum to  $1e - 5$ , which is updated by the Cosine Annealing scheme.

For each model, we calculate the average values of PSNR and SSIM on five datasets. With the same 93.3% memory savings, the proposed NAFNet with UNet— outperforms the baseline model with 0.398 in PSNR and 0.002 in SSIM.



**Table 6:** Models and their performance in image super resolution. SC is the abbreviation for skip connection.

Model	Set5 [3]		Set14 [37]		B100 [19]		Urban100 [13]		Manga109 [20]	
	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
NAFNet [5]	37.74	0.9603	33.47	0.9179	32.16	0.8997	31.98	0.9272	37.96	0.9770
NAFNet w/o SC	36.89	0.9588	32.33	0.9002	31.11	0.8798	31.01	0.9132	36.81	0.9663
NAFNet with UNet—	38.23	0.9611	33.96	0.9207	32.73	0.9017	32.45	0.9321	37.93	0.9768

**Fig. 6:** Visualization results of the proposed NAFNet [5] with UNet— on image super resolution.

#### 4.4 Image matting

Image matting is a fundamental computer vision task which aims to predict an alpha matte to precisely cut out an image region. It estimates the foreground, background, and alpha matte from a single image, but has many applications in image and video editing, virtual reality, augmented reality, entertainment.

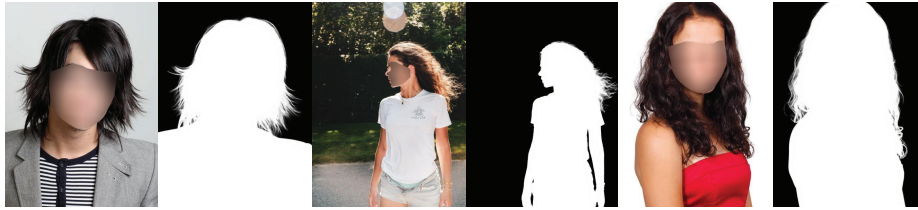
Unlike above image restoration tasks, for image matting task we use MSCAN<sub>tiny</sub> [10] as a new backbone of the encoder.

We apply our proposed method to obtain MSCAN<sub>tiny</sub> with UNet—. We use P3M-10K [16] as the training dataset. It consists of about 10,000 high resolution face-blurred portrait images which are carefully collected and filtered from a huge number of images with diverse foregrounds, backgrounds and postures. In addition, face obfuscation is used as a privacy protection technique to remove identifiable face information while retaining fine details such as hairs. We use three commonly used datasets including P3M-500-P [16], P3M-500-NP [16] and RealWorldPortrait-636 [36] as the test dataset.

We use SAD and MSE metrics to evaluate models on the whole image and the unknown region defined by trimap images. SAD and MSE refer to the sum

**Table 7:** Models and their performance in image matting. SC is the abbreviation for skip connection. The unit represented by \* is  $1e-3$ .

Model	$M_{sc}$ (MB)	MACs (G)	Params (MB)	Dataset	whole image		Unknown	
					MSE*	SAD	MSE*	SAD
MSCAN <sub>tiny</sub> [10]	2.56	1.57	3.95	RWP-636 [36]	14.107	7.519	63.723	5.431
				P3M-500-NP [16]	2.369	1.619	14.926	1.296
				P3M-500-P [16]	2.998	1.736	18.735	1.295
MSCAN <sub>tiny</sub> w/o SC	0.00	1.55	3.95	RWP-636 [36]	12.423	5.925	60.345	3.112
				P3M-500-NP [16]	1.645	1.321	13.994	1.076
				P3M-500-P [16]	2.386	1.184	16.842	1.130
MSCAN <sub>tiny</sub> with UNet--	0.14	1.71	4.23	RWP-636 [36]	14.835	7.611	64.192	5.587
				P3M-500-NP [16]	2.391	1.689	15.164	1.325
				P3M-500-P [16]	3.042	1.728	19.236	1.313

**Fig. 7:** Visualization results of the proposed MSCAN<sub>tiny</sub> [10] with UNet-- on image matting.

of absolute differences and the mean squared error between the ground-truth alpha matte and the predicted alpha matte, respectively.

For each model, we calculate the average values of MSE and SAD on three datasets. With the 94.5% memory savings, the proposed MSCAN<sub>tiny</sub> with UNet-- outperforms the baseline model MSCAN<sub>tiny</sub> with 0.246 (whole image), 0.402 (unknown region) in MSE and 0.051 (whole image), 0.067 (unknown region) in SSIM, respectively. Some visualization results are shown in Fig. 7.

## 5 Conclusion

We have presented an effective method to reduce the memory demand for skip-connections in U-Net and its variants. In addition, we have obtained more enriched information based on the feature map transmitted by skip-connection. Applying the method, we construct a memory-efficient and feature-enhanced network architecture, UNet--. It is shown that the approach achieves better or comparable results compared to the models with common skip-connections on various kinds datasets for different applications, including image denoising, image deblurring, image super resolution, and image matting.

## References

1. Abdelhamed, A., Lin, S., Brown, M.S.: A high-quality denoising dataset for smartphone cameras. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 1692–1700 (April 2018), <https://www.microsoft.com/en-us/research/publication/high-quality-denoising-dataset-smartphone-cameras/>
2. Agustsson, E., Timofte, R.: Ntire 2017 challenge on single image super-resolution: Dataset and study. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW). pp. 1122–1131 (2017). <https://doi.org/10.1109/CVPRW.2017.150>
3. Bevilacqua, M., Roumy, A., Guillemot, C.M., Alberi-Morel, M.L.: Low-complexity single image super-resolution based on nonnegative neighbor embedding. In: British Machine Vision Conference (2012)
4. Bolya, D., Zhou, C., Xiao, F., Lee, Y.J.: Yolact: Real-time instance segmentation. In: 2019 IEEE/CVF International Conference on Computer Vision (ICCV). pp. 9156–9165 (2019). <https://doi.org/10.1109/ICCV.2019.00925>
5. Chen, L., Chu, X., Zhang, X., Sun, J.: Simple baselines for image restoration. arXiv preprint arXiv:2204.04676 (2022)
6. Cheng, T., Wang, X., Chen, S., Zhang, W., Zhang, Q., Huang, C., Zhang, Z., Liu, W.: Sparse instance activation for real-time instance segmentation. In: 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 4423–4432 (2022). <https://doi.org/10.1109/CVPR52688.2022.00439>
7. Ding, X., Zhang, X., Ma, N., Han, J., Ding, G., Sun, J.: Repvgg: Making vgg-style convnets great again. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 13733–13742 (2021)
8. Du, Z., Liu, D., Liu, J., Tang, J., Wu, G., Fu, L.: Fast and memory-efficient network towards efficient image super-resolution. In: 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW). pp. 852–861 (2022). <https://doi.org/10.1109/CVPRW56347.2022.00101>
9. Gao, S.H., Cheng, M.M., Zhao, K., Zhang, X.Y., Yang, M.H., Torr, P.: Res2net: A new multi-scale backbone architecture. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **43**(2), 652–662 (Feb 2021). <https://doi.org/10.1109/tpami.2019.2938758>, <http://dx.doi.org/10.1109/TPAMI.2019.2938758>
10. Guo, M.H., Lu, C., Hou, Q., Liu, Z., Cheng, M.M., Hu, S.: Segnext: Rethinking convolutional attention design for semantic segmentation. *ArXiv abs/2209.08575* (2022), <https://api.semanticscholar.org/CorpusID:252367800>
11. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 770–778 (2016)
12. Huang, H., Lin, L., Tong, R., Hu, H., Zhang, Q., Iwamoto, Y., Han, X., Chen, Y.W., Wu, J.: Unet 3+: A full-scale connected unet for medical image segmentation (2020)
13. Huang, J.B., Singh, A., Ahuja, N.: Single image super-resolution from transformed self-exemplars. *IEEE* (2015)
14. Ibtehaz, N., Kihara, D.: Acc-unet: A completely convolutional unet model for the 2020s. In: Medical Image Computing and Computer Assisted Intervention. vol. 14222 (2023)
15. Li, C., Tan, Y., Chen, W., Luo, X., Gao, Y., Jia, X., Wang, Z.: Attention unet++: A nested attention-aware u-net for liver ct image segmentation. In: 2020 IEEE International Conference on Image Processing (ICIP). pp. 345–349 (2020). <https://doi.org/10.1109/ICIP40778.2020.9190761>

16. Li, J., Ma, S., Zhang, J., Tao, D.: Privacy-preserving portrait matting. In: Proceedings of the 29th ACM International Conference on Multimedia. p. 3501–3509. MM '21, Association for Computing Machinery, New York, NY, USA (2021). <https://doi.org/10.1145/3474085.3475512>, <https://doi.org/10.1145/3474085.3475512>
17. Lin, T.Y., Dollár, P., Girshick, R., He, K., Hariharan, B., Belongie, S.: Feature pyramid networks for object detection. IEEE Computer Society (2017)
18. Loshchilov, I., Hutter, F.: Sgdr: Stochastic gradient descent with restarts (08 2016)
19. Martin, D., Fowlkes, C., Tal, D., Malik, J.: A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. IEEE (2002)
20. Matsui, Y., Ito, K., Aramaki, Y., Fujimoto, A., Ogawa, T., Yamasaki, T., Aizawa, K.: Sketch-based manga retrieval using manga109 dataset. Multimedia Tools and Applications (2017)
21. Nah, S., Kim, T.H., Lee, K.M.: Deep multi-scale convolutional neural network for dynamic scene deblurring. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 257–265 (2017). <https://doi.org/10.1109/CVPR.2017.35>
22. Oktay, O., Schlemper, J., Folgoc, L.L., Lee, M., Heinrich, M., Misawa, K., Mori, K., McDonagh, S., Hammerla, N.Y., Kainz, B., Glocker, B., Rueckert, D.: Attention u-net: Learning where to look for the pancreas. In: Medical Imaging with Deep Learning (2018)
23. Ronneberger, O., Fischer, P., Brox, T.: U-net: Convolutional networks for biomedical image segmentation. In: International Conference on Medical Image Computing and Computer-Assisted Intervention (2015)
24. Shelhamer, E., Long, J., Darrell, T.: Fully convolutional networks for semantic segmentation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. p. 3431–3440 (2015)
25. Singh, G., Alser, M., Denolf, K., Firtina, C., Khodamoradi, A., Cavlak, M.B., Corporaal, H., Mutlu, O.: Rubicon: a framework for designing efficient deep learning-based genomic basecallers. *Genome Biology* **25**, 49 (2024)
26. Sun, G., Pan, Y., Kong, W., Xu, Z., Ma, J., Racharak, T., Nguyen, L.M., Xin, J.: Da-transunet: Integrating spatial and channel dual attention with transformer u-net for medical image segmentation (2023)
27. Szegedy, C., Ioffe, S., Vanhoucke, V., Alemi, A.: Inception-v4, inception-resnet and the impact of residual connections on learning (2016)
28. Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., Wojna, Z.: Rethinking the inception architecture for computer vision. IEEE pp. 2818–2826 (2016)
29. Vasu, P.K.A., Gabriel, J., Zhu, J., Tuzel, O., Ranjan, A.: Fastvit: A fast hybrid vision transformer using structural reparameterization. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (2023)
30. Vasu, P.K.A., Gabriel, J., Zhu, J., Tuzel, O., Ranjan, A.: An improved one millisecond mobile backbone. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (2023)
31. Wang, H., Cao, P., Wang, J., Zaiane, O.R.: Uctransnet: Rethinking the skip connections in u-net from a channel-wise perspective with transformer. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 36, pp. 2441–2449 (2022)
32. Weng, O., Marcano, G., Loncar, V., Khodamoradi, A., Sheybani, N., Meza, A., Koushanfar, F., Denolf, K., Duarte, J.M., Kastner, R.: Tailor: Altering skip connections for resource-efficient inference. *ACM Transactions on Reconfigurable Technology and Systems* **17**, 1–23 (2024)

33. Woo, S., Debnath, S., Hu, R., Chen, X., Liu, Z., Kweon, I.S., Xie, S.: Convnext v2: Co-designing and scaling convnets with masked autoencoders. arXiv preprint arXiv:2301.00808 (2023)
34. Xie, S., Girshick, R., Dollár, P., Tu, Z., He, K.: Aggregated residual transformations for deep neural networks (2017)
35. Yang, X., Li, Z., Guo, Y., Zhou, D.: Dcu-net: a deformable convolutional neural network based on cascade u-net for retinal vessel segmentation. *Multimedia Tools and Applications* **81**, 15593–15607 (05 2022). <https://doi.org/10.1007/s11042-022-12418-w>
36. Yu, Q., Zhang, J., Zhang, H., Wang, Y., Lin, Z., Xu, N., Bai, Y., Yuille, A.: Mask guided matting via progressive refinement network. In: 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 1154–1163 (2021). <https://doi.org/10.1109/CVPR46437.2021.00121>
37. Zeyde, R., Elad, M., Protter, M.: On single image scale-up using sparse-representations. In: International Conference on Curves and Surfaces (2012)
38. Zhang, H., Wu, C., Zhang, Z., Zhu, Y., Lin, H., Zhang, Z., Sun, Y., He, T., Mueller, J., Manmatha, R., Li, M., Smola, A.: Resnest: Split-attention networks. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 2736–2746 (2020)
39. Zhang, Y., Sun, Y., Liu, S.: Deformable and residual convolutional network for image super-resolution. *Applied Intelligence* **52**, 295–304 (Jan 2022)
40. Zhang, Y., Li, D., Shi, X., He, D., Song, K., Wang, X., Qin, H., Li, H.: Kbnnet: Kernel basis network for image restoration. arXiv preprint arXiv:2303.02881 (2023)
41. Zhou, S.K., Greenspan, H., Davatzikos, C., Duncan, J.S., Van Ginneken, B., Madabhushi, A., Prince, J.L., Rueckert, D., Summers, R.M.: A review of deep learning in medical imaging: Imaging traits, technology trends, case studies with progress highlights, and future promises. *Proceedings of the IEEE* **109**(5), 820–838 (May 2021). <https://doi.org/10.1109/jproc.2021.3054390>, <http://dx.doi.org/10.1109/JPROC.2021.3054390>
42. Zhou, W., Bovik, A., Sheikh, H., Simoncelli, E.: Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing* **13**(4), 600–612 (2004). <https://doi.org/10.1109/TIP.2003.819861>
43. Zhou, Z., Siddiquee, M.M.R., Tajbakhsh, N., Liang, J.: Unet++: Redesigning skip connections to exploit multiscale features in image segmentation. *Journal of IEEE Transactions on Medical Imaging* **39**(2), 1856–1867 (2019)
44. Özgün Çiçek, Abdulkadir, A., Lienkamp, S.S., Brox, T., Ronneberger, O.: 3d u-net: Learning dense volumetric segmentation from sparse annotation. Springer, Cham (2016)