A Simple Transformer-style Network for Lightweight Image Super-resolution

Garas Gendy¹, Nabil Sabor², Jingchao Hou¹, Guanghui He¹,∗
¹Micro-Nano Electronics Department, Shanghai Jiao Tong University, Shanghai 200240, China.
²Electrical Engineering Department, Faculty of Engineering, Assiut University, Assiut 71516, Egypt
garasgaras@yahoo.com, nabil_sabor@aun.edu.eg, {jingchaohou, guanghui.he}@sjtu.edu.cn

Abstract

The task of single image super resolution (SISR) has taken much attention in the last few years due to the wide range of real-world applications. However, most of the recently developed methods are computationally expensive and need much more memory. To solve this issue, we propose a simple Transformer-style network (STSN) for the image super resolution (SR) task. The idea of this method is based on using convolutional modulation (Conv2Former), which is a very simple block with a linearly compared to quadratically as in Transformers. This Conv2Former is simplified the self-attention mechanism based on utilizing only convolutions and Hadamard product. Also, the original Conv2Former is further improved to be able to extract local features, which is helpful for SR task. Based on this Conv2Former and multi-layer perceptron (MLP), we propose a convolutional modulation block (Conv2FormerB) which is similar to the Transformers block. Based on this Conv2FormerB, 3 × 3 convolution and enhanced spatial attention (ESA) block, an STSN is designed for the SISR task. This STSN achieved good results in multiple SR benchmarks. Finally, our STSN model attained 5.6 × faster run time compared to LWSwinIR.

1. Introduction

The SISR is a fundamental task of the computer vision domain. This SISR task focuses on generating an output high-resolution (HR) image corresponding to the low-resolution (LR) input one [3, 13, 41]. There are different criteria to classify this SISR task based on degradation, model parameters, the use of deep learning, etc. For example, the models are classified based on parameters into the classical SR model [3, 41], lightweight SR models [13], and mobile scale models [15]. In addition, the models can also classify conventional methods [45] and deep learning-based methods [3, 13, 41].

Recently, the model based on deep learning is taken much attention to solve the SISR task. Dong et al. [11] developed the first deep learning model for the SISR task. However, this model is a very shallow and is not able for extracting more discriminative features. Then, the authors in [29] introduced residual learning to the task of image super-resolution, which helps to increase the model depth to hundreds of layers. In addition, methods-based attention mechanism are widely used due to the ability of attention mechanism for extracting non-local features [10, 49]. Finally, the transformer-based model is successfully used to solve the SISR task [9, 25, 28, 44].

For the transformer-based model, the initial method to use Swin Transformer is made by Liang [28] (SwinIR) to solve the SR task. This SwinIR model is based on using residual Swin Transformer blocks (RSTB) to extract deep feature, which takes benefits to form non-local features and residual learning. After that, an efficient transform is suggested [34] based on designing an efficient Transformer model. Despite the success of these transformer-based models in this task, these models have large computational cost issues for applications that require low latency due to the process of computing self-attention.

In this paper, we tried to solve the Transformer-based model’s problem for solving the SISR task. So, we propose a simple Transformer-style network for image super-resolution (STSN). This STSN model is based on using the original convolutional modulation, but the model is improved to extract local features. This is done by designing the convolutional modulation block (Conv2Former) layer by introducing 3 × 3 instead of 1 × 1 for local feature extraction. Then, a convolutional modulation block (Conv2FormerB) is built based on using Conv2Former and multi-layer perceptron (MLP). Afterward, a convolutional modulation group (Conv2FormerGroup) is designed based on Conv2FormerB, 3 × 3 convolution, and enhanced spatial attention (ESA) block. Finally, the STSN is built based on using Conv2FormerGroup for deep feature extraction.

The paper contribution can be summarized as the following:

- We propose Conv2FormerB, which works as a main block for the SR model, in which its computation rises
linearly rather than quadratically as in Transformer.

- Based on using the Conv2FormerB as the main block, an STSN model is built for the image SR task.
- The proposed method attained the state-of-the-art on the SR benchmark in run time with a good performance. Also, an ablation study is performed to indicate the impact of each model component.

2. Related Work

In this related work section, we will discuss the work related, including two types: classical SR models and lightweight SR models.

2.1. Classical SR

For the classical SR models, these models considered as traditional models include the enhanced deep super-resolution network (EDSR) [29] that is based on using residual learning to improve the SR performance. After that, this EDSR is a further improved residual dense network [50] based on using the dense connection. The deep back-projection networks (DBPN) [16] is introduced based on exploiting iterative up- and down-sampling layers. Then, the ODE-inspired network design model [17] is developed based on using the ordinary differential equation (ODE). Also, based on a graph neural network (GNN), a cross-scale internal graph neural network (IGNN) [54] is developed. However, the recent models that solve the classical SR task are based on Transformer and attention mechanism.

The attention-based models show strong performance in solving the SISR task based on finding non-local features. One of the starting models to use this attention is the residual channel attention network (RCAN) [49] based on using the channel attention mechanism. After that, the second order attention (SAN) [10] is developed using the second-order channel attention (SOCA), which can adaptively rescale the channel-wise features. For the Transformer-based image SR, the initial model is developed based on Swin Transformer [33] to solve the image restoration task in SwinIR [28]. Afterward, the RCAN [49] is further improved [50] based on finding a proper training strategy and minimal changing in the architecture.

Moreover, a hybrid attention transformer (HAT) [7] is developed based on using overlapping cross-attention modules for improving the interaction between neighboring window features. An efficient long-range attention network (ELAN) [48] is developed by Zhang et al. based on calculating self-attention (SA) on non-overlapped feature groups. In [44], a hybrid SR network of CNNs and transformer is introduced based on using CNNs for captioning local features and Transformers to capture long-range multi-scale dependencies. A recursively defined residual network [36] is developed based on the effective use of the attention blocks. Also, a cross aggregation Transformer (CAT) [8] is introduced based on using rectangle-window self-attention (Rwin-SA) that uses parallel horizontal and vertical rectangle window attention in different heads for expanding the attention area and aggregating the features cross different windows. In [47], an attention retractable Transformer (ART) model is developed based on using both dense and sparse attention modules which permit the interaction of tokens from sparse areas for providing a wider receptive field.

2.2. Lightweight SR

For the lightweight SR model, there is strong progress in using CNNs for solving the SR task based on the low computational cost of the convolution operation. For example, the information distillation network (IDN) [21] is developed based on using the distillation of the feature maps. Then, this IDN is further improved [20, 31] based on improving the distillation task. This model is based on extracting feature channels based on the degree of channel redundancy. A hybrid pixel-unshuffled network (HPUN) [39] is suggested by Sun et al. based on using pixel-unshuffled operation for downsampling the input features and utilizing grouped convolution for decreasing the channels. Also, attention in an attention network ($A^2 N$) [6] is developed based on the idea that not all feature maps are helpful to the model.

In addition, Yang et al. [43] developed a feature similarity ranking algorithm image SR task. An efficient non-local contrastive attention (ENLCA) [42] is introduced based on finding long-range dependencies and leveraging more relevant non-local features. In addition, the pixel attention module is further improved [52] based on reducing the model parameters and producing better performance. Then, a blueprint separable residual network (BSRN) [27] is introduced based on designing two blocks one takes the place of the redundant convolution operation. Also, Gendy et al. developed a balanced spatial feature distillation and pyramid attention (BSPAN) [14] for lightweight SR task.

Moreover, the pixel attention module is further improved [53] based on reducing the model parameters and producing better performance. After that, a residual local feature network (RLFN) [23] is developed by Kong et al. based on using three convolutional layers to learn residual for simplifying the feature aggregation. Afterward, many methods are based on using the Transformers are developed [9, 25, 38]. Then, a cross-receptive focused inference network (CFIN) [25] is developed based on using a hybrid model of CNNs and a Transformer.

In addition, N-Gram context is developed for the image super-resolution task based on using Transformer in N-Gram in the Swin Transformers network [9]. In [38], a wavelet-based Transformer for image super-resolution (WTIR) is introduced, which is able to implicitly mine...
the self-similarity of image patches on the wavelet domain. Even though these Transformer-based models attained good performance, these models need a long time for inference, which limit their use in some applications. Based on this limitation of these models, we tried to design a model with linear computational complexity for a fast-processing.

3. Proposed Model.

Our model is built based on using three stages of shadow and deep feature extraction modules and the image reconstruction module. A traditional convolution layer is used for shallow feature extraction. After that, we designed the deep feature extraction based on using the convolutional modulation group (Conv2FormerGroup). Next, we will discuss the details of each block.

3.1. Convolutional Modulation Group (Conv2FormerGroup)

The Conv2FormerGroup is built based on stack $n$ Conv2FormerB blocks with $3 \times 3$ convolution and ESA [32]. Assuming the input feature $F_{i-1}$, the functions of Conv2FormerGroup, then function of this block can be represented as:

$$ F_i = H_{\text{Conv2FormerB}_i}(F_{i-1}), \quad i = 1, 2, ..., n $$

where $H_{\text{Conv2FormerB}_i}$ is the function of $i^{th}$ Conv2FormerB, $H_{\text{Conv3}}$ is the function of $3 \times 3$ convolution, and $H_{\text{ESA}}$ is the function of the ESA layer. This $3 \times 3$ convolution is utilized for extracting the local features. Also, $F_{\text{ESA}}$ is the output feature map of the Conv2FormerGroup. Therefore, the operation of the $g^{th}$ Conv2FormerGroup can be represented as:

$$ F_g = H_{\text{CFG}}(F_{g-1}), $$

where $H_{\text{CFG}}$ is the function of the Conv2FormerGroup and $F_g$ the output of $g^{th}$ layer of Conv2FormerGroup.

3.2. Convolutional Modulation Block (Conv2FormerB)

The Conv2FormerB is designed similarly to the Transformer block of LayerNorm and convolutional modulation (Conv2Former) layer and multi-layer perceptron (MLP). The function of $i^{th}$ Conv2FormerB block ($H_{\text{Conv2Former}_i}$) is defined as follows:

$$ F_{\text{LN}-1} = H_{\text{LN}}(F_{i-1}), $$

$$ F_{\text{CF}} = H_{\text{Conv2Former}}(F_{\text{LN}-1}) + F_{i-1}, $$

$$ F_{\text{LN}-2} = H_{\text{LN}}(F_{\text{CF}}), $$

$$ F_i = F_{\text{MLP}}(F_{\text{LN}-2}) + F_{\text{CF}}, $$

where $F_{\text{LN}-1}$, $F_{\text{CF}}$, $F_i$ are outputs of of LayerNorm layer, Conv2Former layer, and the output feature, respectively.
Also, $H_{LN}$, $H_{Conv2Former}$, and $F_{MLP}$ are the functions of the Layernorm layer, Conv2Former layer, and the MLP layer, respectively. The operation of the $H_{Conv2Former}$ will be explained in the next section.

### 3.3. Convolutional Modulation (Conv2Former) Layer

The convolutional modulation (Conv2Former) layer [18] is first developed for improving the traditional self-attention. However, this Conv2Former layer has some limitations, such as it cannot extract local features, which makes it not helpful for image SR task. To solve this problem, we further replaced the $1 \times 1$ with a traditional $3 \times 3$ convolution, so the model can extract local features. Given an input feature map $F_{LN-1} \in \mathbb{R}^{H \times W \times C}$, the Conv2Former is designed using two branches. The first one is built using only one pointwise convolution, and the second is built using another pointwise convolution followed by depth-wise convolution with a kernel size of $k \times k$. Then, these two branches are multiplied using the Hadamard product. We can express the function of this operation as follows:

$$F_{DC} = H_{DC_{\text{Conv3x3}}}(H_{\text{Conv1}}(F_{LN-1})), \quad (9)$$

$$F_{\text{conv1}} = H_{\text{Conv1}}(F_{LN-1}), \quad (10)$$

$$F_{\text{Conv2Former}} = H_{\text{Conv1}}(F_{DC} \odot F_{\text{conv1}}), \quad (11)$$

where $\odot$ represents the Hadamard product. Also, $H_{\text{Conv1}}$ defines the pointwise convolution layer. $H_{DC_{\text{Conv3x3}}}$ represents a depthwise convolution with kernel size of $3 \times 3$. $F_{DC}$, $F_{\text{conv1}}$, $F_{\text{Conv2Former}}$ are the outputs of the depthwise convolution, pointwise convolution, and the output of the Conv2Former layer. Therefore, the above operations of the Conv2Former can be defined as follows:

$$F_{\text{Conv2Former}} = H_{\text{Conv2Former}}(F_{LN-1}), \quad (12)$$

where $H_{\text{Conv2Former}}$ is the function of Conv2Former.

As proved in ref. [18], the computational complexity of convolution modulation is proportional linearly, the complexity of our model rises linearly because it depends on convolution modulation instead of the similarity score matrix in self-attention in the Transformer.

### 3.4. The simple Transformer-style network (STSN) Framework

Our STSN framework is built based on three modules of shallow feature extraction, deep feature extraction, and image reconstruction, as shown in Fig. 1. The shallow feature extraction is designed using $3 \times 3$ convolution ($H_{\text{conv3}}$) to extract coarse features ($F_0$) from the LR input image. So, we can define this module as:

$$F_0 = H_{\text{conv3}}(x) \quad (13)$$

Then, the deep feature extraction is made using $m$ layers of Conv2FormerGroup ($H_{\text{CFG}}$).

$$F_g = H_{\text{CFG}}_g(F_{g-1}), \quad g = 1, 2, \ldots, m \quad (14)$$

where $F_g$ defines the output of the $g$ layer of the Conv2FormerGroup. Following that, both the coarse feature map ($F_0$) and the deep feature Conv2FormerGroup ($F_g; g = 1, 2, \ldots, m$) are concatenated. Then, we included both $3 \times 3$ and $1 \times 1$ to smooth the aggregated features as fellows:

$$F_{\text{Comb}} = H_{\text{Conv}}(H_{\text{ConCat}}(F_0, F_1, F_2, \ldots, F_m)), \quad (15)$$

where the $H_{\text{ConCat}}$ is mean to concatenate the the channel dimension. Also, $H_{\text{Conv}}$ defines $1 \times 1$ convolution next by a $3 \times 3$ convolution, $F_{\text{Comb}}$ represents the overall feature map form combing both coarse and deep features. Finally, the output SR image is generated using reconstruction modules as follows:

$$y = H_{\text{recont}}(F_{\text{Comb}} + F_0), \quad (16)$$

where $H_{\text{recont}}$ defines the reconstruction function, which contains both $3 \times 3$ convolution and Sup-pixel upsampling. Finally, $y$ represents the output of the model.

### 4. Experiment

#### 4.1. Benchmarks

For the training section, the DIV2K [1] dataset is utilized for training our method, and an down-sampling the HR image using the bicubic down-sampling to generate the LR image. We tested the model using a benchmark of 5 datasets, including Set5 (5 images) [5], Set14 (14 images) [46], B100 (100 images) [4], Urban100 (100 images) [19], and Manga109 (109 images) [35]. Finally, the PSNR and the structural similarity index (SSIM) [40] are used for model evaluation based on using the $Y$ channel.

#### 4.2. Implementation Details

We set the patch size to 96, 144, 192 for scales the $\times 2, \times 3,$ and $\times 4$, respectively. Also, batch sizes of 32 are used for training. In addition, 90, 180, and 270 degrees of random rotation and horizontal flipping are used as augmentation methods for the input images. Then, the number of Conv2FormerGroup blocks ($n$) is set to 4 for STSN models. Additionally, the number of the Conv2FormerB ($n$) in the Conv2FormerGroup is empirically set to 4 for STSN models to balance the performance and computation. Moreover, the number of features is set to 50 for the STSN. The ADAM optimizer [22] is utilized with $\beta_1 = 0.9$, $\beta_2 = 0.99$ and $\epsilon = 1e^{-8}$. Also, the learning rate begins with $5 \times 10^{-4}$.
for the STSN, and half every 200 epochs. The $L_1$ loss function is used to train the model for 1000 epochs. In addition, the warm-start strategy [23] is used for the STSN but not used for the ablation study. Finally, we built the model using the PyTorch [37] framework and trained based on using Nvidia 2080 Ti GPUs.

### 4.3. Comparison with State-of-the-Art Models

This section compares our methods with 8 state-of-the-art lightweight images SR methods such as SRCNN [11], FSRCNN [12], CARN [2], LapSRN [24], IDN [21], IMDN [20], RFDN [31], $A^2N$ [6], ELAN-light [48], and LWSwinIR [28]. We compared our model with these methods in three factors quantitative, qualitative, and model size analyses.

#### 4.3.1 Quantitative Evaluations

To show the quantitative result of our model, five test datasets are used to compare our model with other state-of-the-art models, as illustrated in Table 1. In this case, the STSN model is used for different scale factors. It is clear from the table that our model achieved better performance compared to $A^2N$ [6], ELAN-light [48], and LWSwinIR [28]. However, our model achieved a much faster run time due to the simple model design. For instance, the STSN model improved from 38.06 dB and 32.22 dB to 38.19 dB and 32.30 dB compared to $A^2N$ at the scale of $\times2$. Also, the STSN model improved from 34.00 dB to 34.11 dB compared to ELAN-light at the scale of $\times3$ in the Manga109 dataset. In addition, the SSIM for our model at the scale of $\times4$ is improved from 0.8976 and 0.7858 to 0.8982 and 0.7860 compared to LWSwinIR for Set5 and Set14, respectively.

#### 4.3.2 Qualitative Evaluations

To show the efficient performance of our model, we made a comparison with a state-of-the-art model in the qualitative result. As indicated in Fig. 2, our model shows good results compared to the other models. For example, for img 070 in the Urban100 dataset, the details are much clearer compared to the LWSwinIR, which represents the state-of-the-art result. In addition, for img 074, the details of the lines are much more clearer compared to the other methods. It is clear from the result that our model can achieve good visual quality.

#### 4.3.3 Model Size Analysis

We tried to make a comparison with other state-of-the-art models in the case of parameters, Multi-Adds, and runtime.
Figure 2. Urban100 Dataset Visual Comparison at × 4 SR.

as shown in Table 1. The table indicates that our model has many similarities in parameter and Multi-Adds compared to LWSwinIR [28]; however, our model has a much faster run time. In addition, our model has lower parameters, Multi-Adds, than A²N [6] for all the SR scales. For example, our model has 17 % and 25 % fewer in the number of parameters, and Multi-Adds compared to the A²N with much better performance. Moreover, for the run time, our model is 5.6 × faster runtime than to LWSwinIR, with mostly similar performance. This indicates that our model is efficient for the application that needs faster models.

4.4. Ablation Study

In our ablation, the STSN model is used before using the warm-start strategy to save time. Also, we used the model at scale of × 2. The study aim is to study the impact of factors, the impact of some modules in the Conv2Former, the impact of modules in the Conv2FormerB, the impact of modules in the Conv2FormerGroup, and the impact of the warm-Start Strategy.

4.4.1 Ablation Study in the Conv2Former Block

The impact of using 3 × 3 instead of 1 × 1. To illustrate why the 1 × 1 conv of the original Conv2Former is replaced by 3 × 3 conv, the improved Conv2Former and the original one are used independently in the proposed model, as indicated in Fig. 3a (Model 1). The results are listed in Table 2, where 1st row represents the results of using improved Conv2Former, and 2nd row represents the results of using the original Conv2Former. The result indicated that the 3 × 3 greatly impacts all test datasets, especially for the Urban100 dataset. This is because the 3 × 3 can extract local features, leading to performance improvement.

The impact of using attention module. In this task, all the contents of the conv2Former block are removed, except of the 3 × 3 to indicate the impact of the attention module, as indicated in Fig. 3b (Model 2). The obtained results are indicated in Table 2, where 1st row represents the results of using the Conv2Former, and 3rd row represents the results without using the attention module. The results show that
the attention module has a big impact on performance. For instance, the PSNR dropped from 33.77 dB to 33.61 dB on the Set5 dataset. So, these results show that the attention module can greatly impacts the performance.

4.4.2 Ablation Study in the Conv2FormerB Block

The impact of using the Conv2Former block. In this task, the conv2Former block is removed from the Conv2FormerB to indicate the impact of the conv2Former on the performance, as indicated in Fig. 3c (Model 3). The obtained results are listed in Table 3, where 1st row represents the results of using Conv2Former, and 2nd row represents the results of model without using the conv2Former layer in conv2FormerB block. The results indicate that this block has an impact on performance. For example, the PSNR dropped from 39.06 dB to 38.70 dB on the Manga109 dataset. So, the conv2Former block in the Conv2FormerB block can greatly impact the performance due to its ability to extract local and non-local features.

The impact of using the MLP block. In this task, the MLP block is not included from the Conv2FormerB to indicate the impact of the MLP on the performance, as indicated in Fig. 3d (Model 4). The obtained results are shown in Table 3, where 1st row represents the results of using the full model, and 3rd row represents the results without using MLP layer in conv2FormerB block. The results show that the MLP block has a significant impact on performance. For instance, the PSNR decreased from 32.58 dB to 32.30 dB on Urban100 dataset. So, the MLP block in the Conv2FormerB block could impact the performance due to its ability to make feature transformations.

4.4.3 Ablation Study in the Conv2FormerGroup Block

The impact of using the ESA block. In this task, the ESA is not included in the Conv2FormerGroup to show the impact of the ESA on the performance, as shown in Fig. 3e (Model 5). The obtained results are indicated in Table 4, where 1st row represents the results of using the full model, and 2nd row represents the results without using ESA layer. The results illustrates that the ESA convolution impacts performance. For example, the PSNR decreased from 38.16 dB to 38.10 dB on the Set5 dataset. So, these
results show that the ESA block could impact the performance due to its ability to use spatial attention to improve the performance.

The impact of using the $3 \times 3$ convolution. In this task, the $3 \times 3$ convolution is removed from the Conv2FormerGroup to show the impact of the $3 \times 3$, as indicated in Fig. 3f (Model 6). The obtained results are shown in Table 4, where 1<sup>st</sup> row represents the results of using the full model, and 3<sup>rd</sup> row represents the results without using $3 \times 3$ convolution. The results show that the $3 \times 3$ convolution impacts performance greatly. For instance, the PSNR decreased from 32.58 dB to 32.44 dB on the Urban100 dataset. So, these results show that the $3 \times 3$ convolution could impact the performance due to its ability to extract local features.

4.4.4 The Ablation Study on Warm-Start Strategy

In this task, the warm-start strategy [23] is used to retrain the model again, starting from the pre-train model on the same scale. The obtained results are indicated in Table 5, where 1<sup>st</sup> row represents the results of using the full model, and 2<sup>nd</sup> row represents the results with this strategy. The results indicate the model performance on PSNR improved from 32.58 dB and 39.06 dB to 32.68 dB and 39.13 dB on the Urban100 and Manga109, respectively. So, these results show that this strategy can impact the performance without any additional parameters and Multi-Adds.

4.5. STSN for NTIRE 2023 Challenge

We took part in NTIRE 2023 Image Super-Resolution Challenge [51], and our model achieved a good result, as shown in Table 6. Our STSN model is changed from the STSN model in the paper; it contains five Conv2FormerGroup blocks containing 4 Conv2FormerB, in which the number of feature maps is set to 150. Also, the channel number of the ESA is set to 32, similar to [32], and we set the RGB range to 255, not to 1, as in the paper. In our training, we used DIV2K and LSDIR [26] to train the model. After that, the model is trained in the following steps. At the starting stage, the model is trained from scratch using the DIV2K and LSDIR [26] datasets, with a patch size of $192 \times 192$. We train our model using a batch size of 16 for 70 epochs. Then, the pre-trained weights are used to train it again for 450 epochs with the same setting based on using the warm-start strategy [23]. In this training, $L_1$ loss function is used with the Adam optimizer. After the previous stage, we trained the model starting from the previous pre-trained weights using the DIV2K and Flickr2K datasets with an learning rate of $5 \times 10^{-5}$ for 200 epochs using $L_1$ loss. Using this model design and training strategy, we got among the best 10 teams of the competition.

5. Conclusion

In this paper, we propose a simple Transformer-style network (STSN) for single image super-resolution (SISR). The STSN is designed to be similar to the Transformers block but with linear complexity. The idea of this method is based on using convolutional modulation (Conv2Former), which is a very simple block with a linearly compared to quadratically in Transformers. This model simplified the self-attention mechanism based on using only convolutions and Hadamard product. Our methods achieved faster run time based on the experimental result in SR models.

Acknowledgment. This work was supported in part by the National Key Research and Development Program of China under Grant 2019YFB2204500, in part by the National Natural Science Foundation of China under Grant 62074097.
References


[25] Yawei Li, Kai Zhang, Jingyun Liang, Jiezhang Cao, Ce Liu, Rui Gong, Yulun Zhang, Hao Tang, Yun Liu, Denis Demanox, et al. Lsdir: A large scale dataset for image restoration. 8


