Attention in Attention Network for Image Super-Resolution

Haoyu Chen
Amazon Web Services
haoyuc@amazon.com

Jinjin Gu
The University of Sydney
jinjin.gu@sydney.edu.au

Zhi Zhang
Amazon Web Services
zhiz@amazon.com

Abstract

Convolutional neural networks have allowed remarkable advances in single image super-resolution (SISR) over the last decade. Among recent advances in SISR, attention mechanisms are crucial for high performance SR models. However, few works really discuss why attention works and how it works. In this work, we attempt to quantify and visualize the static attention mechanisms and show that not all attention modules are equally beneficial. We then propose attention in attention network (A^2N) for highly accurate image SR. Specifically, our A^2N consists of a non-attention branch and a coupling attention branch. Attention dropout module is proposed to generate dynamic attention weights for these two branches based on input features that can suppress unwanted attention adjustments. This allows attention modules to specialize to beneficial examples without otherwise penalties and thus greatly improve the capacity of the attention network with little parameter overhead. Experiments have demonstrated that our model could achieve superior trade-off performances comparing with state-of-the-art lightweight networks. Experiments on local attribution maps [9] also prove attention in attention (A^2) structure can extract features from a wider range. Codes are available at https://github.com/haoyuc/A2N.

1. Introduction

Image super-resolution (SR) is a low-level computer vision problem, which aims at recovering a high-resolution (HR) image from a low-resolution (LR) observation. In recent years, SR methods based on deep convolution neural networks (CNN) have achieved significant success, the performance of the CNN model is constantly growing. Recently, some methods begin to aggregate attention mechanism into the SR model, *e.g.*, channel attention and spatial attention [36, 11, 16, 29, 20]. The introduction of attention mechanism greatly improves the performance of these networks by enhancing the representation capability of static CNNs. Some works [36] believe that attention will enhance high-frequency details, with the assumption that high-frequency information is more valuable in image SR. However, very few works have really proved this assumption, We ask two questions here: 1) What feature will attention layers strengthen? 2) Is attention always effective?

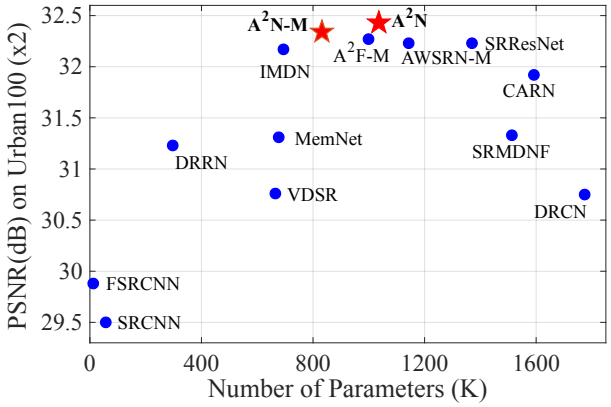

Figure 1: Performance comparison between our A^2N and other state-of-the-art lightweight networks (blue circle) on Urban100 with a scale factor of 2.

In this paper, we first answer the first question by assessing the most valuable areas highlighted by attention. We observed that for a CNN, the attention layers in the shallow stages of the network tend to enhance the feature low-frequency bands, and the subsequent attention generally tends to enhance the high-frequency bands of the feature maps. Then we disabled the attention modules of some layers, and finally get the same performance as using attention on every layer. This shows that not all attention can improve network performance. There are still some inefficient and redundant features. Using attention on all layers is not the most efficient. Based on the above findings, we propose a low-consumption attention in attention (A^2) structure, which is divided into attention branch and non-attention branch. The attention branch is used to reinforce useful information, and the non-attention branch is used to learn information that is ignored by attention. It is critical that we propose an attention dropout module to dynamically allocate the weights of two branches, make full use of the information of the two branches, enhance the high contri-

bution information and suppress the redundant information. Experiments have proved that the A^2 structure is better than the traditional attention structure.

Overall, the contributions of this work are three-fold:

(1) We quantified the effectiveness of attention layers across different stages in neural networks, and propose a valid strategy for pruning attention layers.

(2) We propose an attention in attention block (A^2B), which dynamically produces sum-to-one attentions for its internal branches. One of the internal branches is attention branch, thus our A^2B performs attention in attention operations.

(3) We propose A^2N based on the A^2B , which demonstrate superior performance compared to baseline networks using a similar architecture with small parameter overhead. [9].

2. Related Work

2.1. Deep CNN for SR

Recently, CNN-based methods for single image super-resolution (SISR) have achieved significant promotion. The network design is one of the most important parts in SR problem. Pioneer work of SRCNN [7] first introduced a shallow three-layer convolutional neural network for image SR, which shows superior performance of deep learning. After this work, the network architecture is constantly improving. By introducing residual learning to ease the training difficulty, Kim et al. proposed deeper models VDSR[14] and DRCN [15] with more than 16 layers. In order to learn higher-level features without introducing overwhelming parameters, recursive learning is also introduced into the SR field [31]. DRCN [15] employs a single convolutional layer as the recursive unit for 16 recursions. EDSR [17] achieves significant improvement by removing unnecessary modules (batch normalization) in residual networks. For the sake of fusing low-level and high-level features to provide richer information and details for reconstructing, RDN [38], CARN [2], MemNet [25] and ESRGAN [30] also adopted dense connections in layer-level and block-level. Liu et al. [19] propose a novel residual feature aggregation framework to fully utilize the hierarchical features on the residual branches.

2.2. Attention Mechanism

Attention mechanism in deep learning is similar to the attention mechanism of human vision. It can be viewed as a means of biasing the allocation of available computational resources towards the most informative components of a signal [10]. Attention mechanism usually contains a gating function to generate a feature mask.

Attention mechanism has been applied to many computer vision tasks, such as image captioning [33, 4] and image classification [10, 27]. Wang et al. [28] initially proposed non-local operation for capturing long-range dependencies, it computes the response at a position as a weighted sum of the features at all positions, which brings solid improvement. Hu et al. [10] focus on the channel relationship and propose a novel architectural unit, Squeeze-and-Excitation (SE) block, that adaptively recalibrates channel-wise feature responses. Woo et al. [32] provided a study on the combination of channel and spatial attention, which are in a sequential manner, focusing on 'what' and 'where' respectively.

In recent years, several works proposed to investigate the effect of attention mechanism on low-level vision tasks. [18] first attempt to incorporate non-local operations into a recurrent neural network for image restoration. RNAN [37] proposed residual local and non-local attention blocks in the mask branch in order to obtain non-local mixed attention. Channel attention is another popular way to embed attention mechanism. RCAN [36] exploits the interdependencies among feature channels by generating different attention for each channel-wise feature. Some works utilize both channel attention and non-local attention. SAN [6] performed region-level non-local operations for reducing computational burden, and proposed second-order channel attention by considering second-order statistics of features.

3. Motivation

Given an intermediate feature map $\mathbf{F} \in \mathbb{R}^{C\times H\times W}$, where C, H, and W are the number of channels, height, and width of the features, respectively. Attention mechanism infers an attention map function $\mathbf{M_A}$, where $\mathbf{M_A}(\mathbf{F}) \in \mathbb{R}^{C'\times H'\times W'}$, the size of C', H' and W' depend on the type of attention function. For example, channel attention generates a 1D ($\mathbb{R}^{C\times 1\times 1}$) channel-wise attention vector [10, 36, 32, 23, 11]. Spatial attention generates a 2D ($\mathbb{R}^{1\times H\times W}$) attention mask, [32, 23, 11]. Channel-spatial attention generates 3D ($\mathbb{R}^{C\times H\times W}$) attention map, [37, 39]. We naturally raise two questions: (1) Which part of an image tends to have a higher or lower attention coefficient? (2) Are attention mechanisms always beneficial to SR models?

3.1. Attention Heatmap

Information in the LR space has abundant low-frequency and valuable high-frequency components. Previous work [36] suggests that all features are treated equally without using attention mechanism in the networks, while attention can help the network pay more attention to the high-frequency features. However, to our best knowledge, few works truly prove the above assumptions. To answer the first question, we conduct experiments to understand the

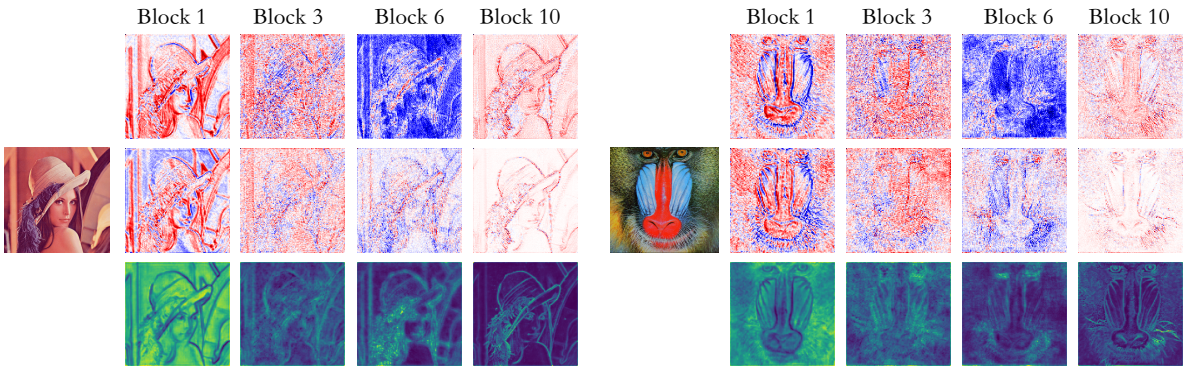

Figure 2: Attention block heatmaps. Due to the limited space, we chose several representative blocks, each columns indicates the first, third, sixth, tenth attention block, respectively. **The first row**: averaged input feature map of attention layers. **The second row**: averaged output feature map of attention layers. **The third row**: averaged attention map. For the first two rows, the white area in the feature map indicates zero values, the red area indicates positive values, and the blue area indicates negative values. For the attention map (the third row), brighter colors represent a higher attention coefficient.

Table 1: The correlation coefficients between the attention map and the output feature of high-pass filters of the corresponding feature map for each attention block .

Attention Block Index		1	2	3	4	5	6	7	8	9	10
Highpass	Laplace	-0.270	-0.082	0.323	0.281	0.296	0.158	-0.098	0.100	0.184	**0.433**
	Scharr	-0.395	-0.122	0.398	0.355	0.412	0.206	-0.079	0.073	0.255	**0.435**
	Sobel	-0.393	-0.153	0.371	0.334	0.379	0.175	-0.109	0.090	0.196	**0.491**

behavior of the attention mechanism in SR. We construct a simple attention-based model, which consists of ten attention blocks. Each attention block uses a channel- and spatial-wise attention layer so that every pixel has an individual attention coefficient. We use the sigmoid function as the gating function so that the attention coefficient can be scaled into $[0, 1]$. We visualize some feature maps and attention maps in Figure 2. Table 1 lists correlation coefficients between the attention map and the high-pass filtering results to the corresponding feature map. Note that this is not a highly accurate method to measure the exact attention response, but our intention is to quantify the relative high-pass correlation across different layers. Based on the observations from Figure 2 and Table 1, we show that attentions learnt at different layers vary a lot with respect to their relative depth in the neural network. For example, the first and tenth attention blocks show opposite responses, indicating attentions favor low-frequency patterns at lower levels and high-frequency patterns at higher levels. The blocks in-between have mixed responses.

3.2. Attention Dropout

Based on the above results, we may be able to maximize the use of attention while minimizing the number of additional parameters. An intuitive idea is to preserve attention layers only at performance-critical layers. However, the above qualitative analysis is not a valid method to quantify the realistic effect of attention layers. To quantitatively measure the effectiveness of attention layers, we propose an attention dropout framework. An attention block, regardless of its type mentioned previously in Section 2.2, can be downgraded to a non-attentional block by simply removing the attention generator operation.

We have conducted a series of experiments with certain attention layers turned off. The results are shown in Table 2, where the first column indicates residual attention blocks that are enabled. For example, $\{1, 2, 3, 4, 5\}$ means attention layers in the first five blocks are residual attention blocks while others are turned off and downgraded to basic residual blocks. The results lead to an interesting result: the relative block depth matters a lot in the decision where to insert attention blocks. Enabling $\{6, 7, 8, 9, 10\}$ blocks with attention is effectively achieving the same PSNR as tuning on every block but with much fewer parameters. This experiment further proves that spending budget on attention uniformly across the network is sub-optimal.

4. Method

Previous models [36, 39, 11] with fixed attention layers have attention maps activated all the time, regardless of im-

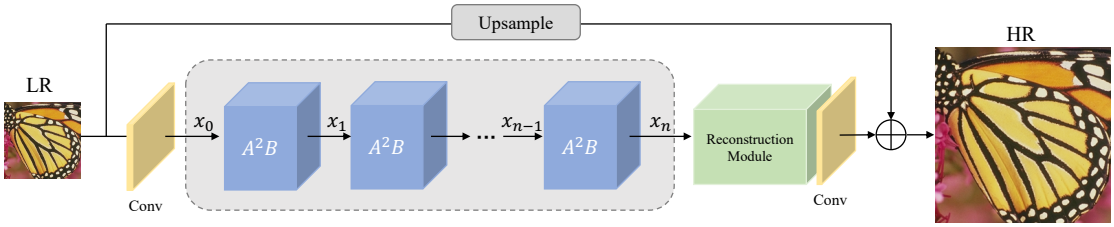

Figure 3: Overview of attention in attention network (A^{2}N).

Table 2: Attention layer importance measurement. The first column indicates the insertion stage of the residual attention blocks.

Attention Block Index	PSNR	# Parameter
All	**28.65**	9.2 M
None	28.60	4.4 M
{1,2,3,4,5}	28.60	6.8 M
{6,7,8,9,10}	**28.65**	6.8 M
{2,4,6,8,10}	28.63	6.8 M

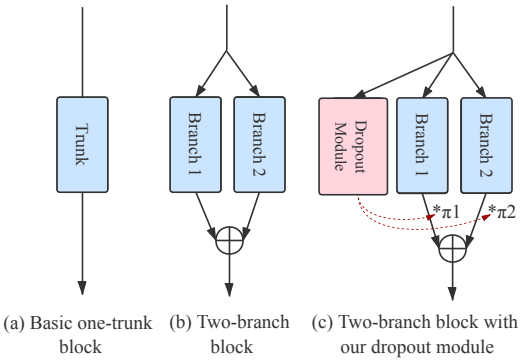

Figure 4: Attention Dropout Module.

age content. We have shown in Sec. 3.2 that the effectiveness of attention layers vary at different locations in the neural network. Our motivation here is to create non-attentional shortcut branches for counter-part attention branches with mixing weights generated dynamically through an additional evaluation module using the same input features as ordinary layers.

4.1. Network Architecture

As shown in Figure 3, the network architecture of our proposed method, consists of three parts: shallow feature extraction, attention in attention block deep feature extraction, and reconstruction module. The input and output image is denoted as I_{LR} and I_{SR}. Follow [17], we use a single convolution layer in shallow feature extraction module. We can then formulate $x_0 = f_{ext}(I_{LR})$, where $f_{ext}(\cdot)$ is a convolution layer with 3×3 kernel size to extract the shallow feature from the input LR image I_{LR}, x_0 is the extracted feature map.

We construct our deep feature extractor as a chained subnetwork using A^{2}B.

$$x_n = f^n_{A^2B}\left(f^{n-1}_{A^2B}\left(\ldots f^0_{A^2B}\left(x_0\right)\ldots\right)\right) \tag{1}$$

where $f_{A^2B}(\cdot)$ denotes our proposed attention in attention block. A^{2}B combines non-attention branch and attention branch with dynamic weights.

After deep feature extraction, we upscale the deep feature x_n via the reconstruction module. In the reconstruction module, we first use nearest-neighbor interpolation for upsampling, then we use a simplified channel-spatial attention layer between two convolution layers. This simplified attention layer only uses one 1×1 convolution and sigmoid function to generate the attention map. We also use global connection, in which a nearest-neighbor interpolation is performed on the input I_{LR}. The final model produces high resolution result by applying the reconstruction signal to upsampled output:

$$I_{SR} = f_{rec}(x_n) + f_{up}(I_{LR}) \tag{2}$$

$f_{rec}(\cdot)$ is the reconstruction module, $f_{up}(\cdot)$ is the bilinear interpolation. I_{SR} is the final SR output.

4.2. Attention in Attention Block (A^{2}B)

We have discussed and dynamic contribution from different attention layers in section 3.2, nevertheless it is infeasible to manually determine the topological structure of attention modules. Inspired by the dynamic kernel [5] which use dynamic convolution to aggregate multiple parallel convolution kernels dynamically based upon their attentions, here we propose a learnable attention dropout module to automatically "dropout" some unimportant attention features and balance the attention branch and non-attention branch. More specifically, each attention dropout module controls the dynamic weighted contribution from the attention and the non-attention branch using weighted summation. As depicted in Figure 4 attention dropout module generates

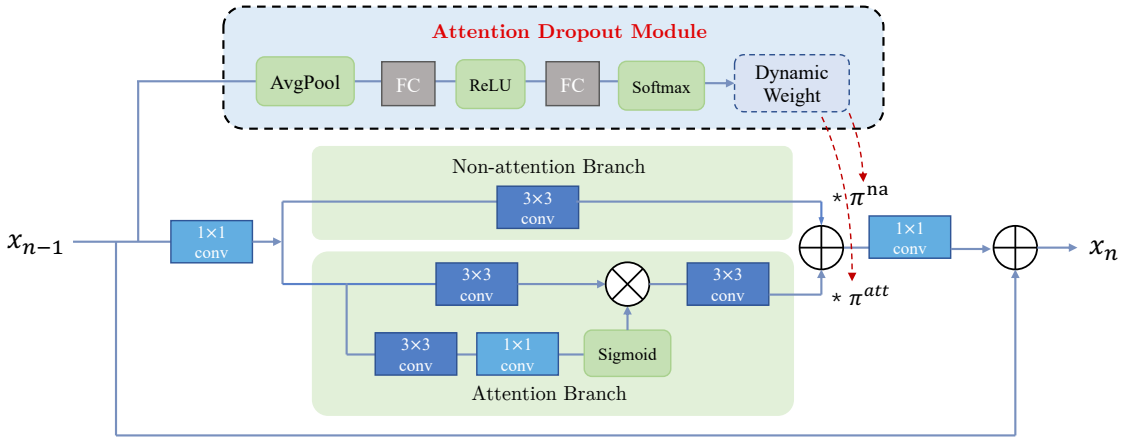

Figure 5: Architecture of the attention in attention block (A^2B).

weights by using the same input feature of its block as two independent branches. Formally, we have:

$$x_{n+1} = f_{1\times1}(\pi_n^{na} \times x_n^{na} + \pi_n^{att} \times x_n^{att}) \quad (3)$$

where x_n^{na} is the output of non-attention branch, and x_n^{att} is the output of the attention branch. $f_{1\times1}(\cdot)$ donates 1 $\times$ 1 kernel convolution. π_{na} and π_{att} are weights of non-attention branch and attention branch respectively, they are computed by the network according to the input feature, instead of two fixed values which are artificially set. To compute the dynamic weights, we have:

$$\pi_n = f_{drop}(x_n) \quad (4)$$

where $f_{drop}(\cdot)$ is the attention dropout module. The attention dropout module can be viewed in detail in Figure 5. It firstly squeezes the input x_{n-1} using global average pooling. The connecting layers consist of two fully connected layers with a ReLU activation. We use global pooling to increase receptive field, which allows attention dropout module to capture features from the whole image, section 5.4 gives experiments to prove this.

As investigated in [5], constraining the dynamic weights can facilitate the learning of attention dropout module. Specifically, we have the sum-to-one constraint $\pi_n^{na} + \pi_n^{att} = 1$. This sum-to-one constraint for the dynamic weights can compress the kernel space. it significantly simplifies the learning of π. Therefore, a softmax function is followed to generate normalized attention weights for two branches.

There are many combinations of different types of attention implementation that can be used in the two branches, in section 5.3, we conduct detailed experiments on it. The overall structure of our proposed attention in attention block in shown in Figure 5. $\otimes$ combines feature and attention map by element-wise multiplication, $\oplus$ computes weighted summation over two branches as Eq. 3.

5. Experiments

In this section, we compare our method with state-of-the-art SISR algorithms on five commonly used benchmark datasets. Besides, we conduct ablation study to validate and analyze the effectiveness of our proposed method.

5.1. Datasets and Metrics

We use DIV2K dataset [1] as our training dataset, which contains 800 training images. The LR images are obtained by the bicubic downsampling of HR images. For testing stage, we use five standard benchmark datasets: Set5[3], Set14[34], B100[21], Urban100[12] and Manga109[22]. The SR results are evaluated by peak signal to noise ratio (PSNR) and the structural similarity index (SSIM) on the Y channel of YCbCr space.

5.2. Implementation Details

Now we specify the implementation details of our proposed A^2N. We design two variants of A^2N, denoted as A^2N and A^2N-M. On non-attention branch, we use 3×3 convolution for A^2N and 1×1 convolution for A^2N-M. For both variants, we set the number of A^2B as 16. Features in A^2B have 40 filters, except for that in the upsampling block, where C = 24. For the training process, data augmentation is performed on the 800 training images, which are randomly rotated by 90°, 180°, 270°, and flipped horizontally.

5.3. Results

To demonstrate the effect of our proposed attention in attention (A^2) structure, we compare our two-branch A^2 structure with a one-trunk structure. The results are shown in Table 4. If we only keep the attention branch, similar to previous models, 28.646 dB PSNR is obtained with 810K parameters. Using our A^2 structure, the performance increased by 0.05 dB with only 200K extra parameters; If we

Table 3: Ablation study: effect of different components. Test on Set14 ($\times$ 4).

Case Index	1	2	3	4	5	6	7	8
non-attention	✓		✓	✓	✓		✓	✓
channel attention	✓	✓			✓	✓		
spatial attention		✓		✓		✓		✓
channel-spatial attention			✓				✓	
A^2					✓	✓	✓	✓
Parameter	787K	1035K	1040K	791K	794K	1042K	1047K	798K
PSNR	28.634	28.649	28.651	28.583	28.600	28.629	28,707	28.642
Gain from A^2	-	-	-	-	-0.034	-0.020	**+0.056**	**+0.059**

Table 4: The effect of attention in attention. A^2N-non-attn-only: no attention branch. A^2N-attn-only: attention branch only. A^2N-Addition: fuse features by addition instead of attention dropout module. A^2N-S: smaller channels. A^2N-M: replacing 3 $\times$3 convolution with 1$\times$1. Test on Set14 ($\times$ 4).

Method	Params	PSNR
A^2N-non-attn-only	208K	28.515
A^2N-attn-only	810K	28.646
A^2N-Addition	1040K	28.651
A^2N-Concatenation	1092K	28.642
A^2N-AdaptiveWeights	1040K	28.648
A^2N-S	678 K	28.651
A^2N-M	843K	28.695
A^2N	1047K	28.707

reduce the number of channels of our method to 32, the A^2 structure performs better with 132K fewer parameters. The results also prove once again that not all attention layers are making positive contributions.

We also show results in Table 3 to evaluate the performance of our A^2 structure on the multi-branch model. Case 1- 4 are two-branch models without the A^2 structure, features from two branches are fused by an addition operation. Case 5 - 8 modify case 1- 4 by applying the A^2 structure. As we can see, for case 1- 4, the combination of non-attention and channel-spatial attention has the best performance. Therefore, we use channel-spatial attention in the attention branch. For case 1- 4, the non-attention branch combine with spatial attention or channel-spatial attention (case 6 and 7) gain more than 0.05 dB by only about 7K parameters cost. Therefore, attention dropout module performs well when used in models without pooling or downsampling layers.

We compare our method with various SR methods of similar model sizes: SRCNN [7], FSRCNN [8], DRRN [24], VDSR [14], MemNet [25], IMDN [13], A^2F-M [29], AWSRN-M [26], SRMDNF [35], CARN [2] and DRCN [15]. Table 5 shows quantitative comparisons for $\times$2, $\times$3, and $\times$4 SR. Note that we only compare models which have a similar number of parameters to our models in this table. Our A^2N can achieve comparable or better results than state-of-the-art methods for all scaling factors. In particular, A^2N-M, which has about 200K fewer parameters than A^2F-M and AWSRN-M, achieves better results than these models on most datasets. For Manga109 ($\times$3), the PSNR of A^2N-M is 0.15 dB higher than the PSNR of AWSRN-M.

Figure 7 shows the visual comparison for upscaling factor $\times$4. We can see that our method achieves better performance than others, it can recover high frequency details more accurately. For image "zebra", our method can restore zebra stripes accurately. For the details of the buildings, in image "78004", the windows are restored more clearly and accurately. For image "img092", only our method can restore the direction of the fringe correctly.

5.4. Discussion

We use the weight prediction in the attention dropout layer to sample the enhanced and suppressed attention signals. Figure 7 shows the two attention maps with the top two weights and the two attention maps with the smallest weights. For each row, the attention maps selected by the highest and second-highest weights are listed on the left, while the right depicts the two attention maps with the smallest weights. As we can see, for attention maps with high weight, the high-frequency / low-frequency regions of the feature map can be accurately located, while the attention maps with lower weight can not found it accurately. These results demonstrate the effectiveness of attention dropout module, which can automatically determine the weights of the two branches based on the input features.

Comparison with Other Fusion Methods. Most SR models fuse features by addition [16] or concatenation [13, 39]. Some methods [15, 29, 26] give adaptive weights to each feature, which means the independent weights will be learned automatically when training the model. To demonstrate the effectiveness of our method, we compare our attention dropout module with other mainstream feature fusion methods: addition, concatenation and adaptive weights. Table 4 shows that within a similar parameter number, our A^2 structure has a considerable improvement over other fusion methods. If we reduce the channel number to 32, even with about 400K less parameter, our A^2N can still obtain a better result than other fusion Methods. It demonstrates that the attention dropout module is a better

Table 5: Quantitative results of state-of-the-art SR methods for all upscaling factors $\times 2$, $\times 3$, and $\times 4$. Red/Blue text: best/second-best among all methods.

Scale	Size Scope	Model	Params	MutiAdds	Set5	Set14	B100	Urban100	Manga109
2	<1,000 K	SRCNN [7]	57K	52.7G	36.66/0.9542	32.45/0.9067	31.36/0.8879	29.50/0.8946	35.60/0.9663
		FSRCANN [8]	12K	6G	37.00/0.9558	32.63/0.9088	31.53/0.8920	29.88/0.9020	36.67/0.9710
		DRRN [24]	297K	6797G	37.74/0.9591	33.23/0.9136	32.05/0.8973	31.23/0.9188	37.92/0.9760
		VDSR [14]	665K	612.6G	37.53/0.9587	33.03/0.9124	31.90/0.8960	30.76/0.9140	37.22/0.9729
		MemNet [25]	677K	2662.4G	37.78/0.9597	33.28/0.9142	32.08/0.8978	31.31/0.9195	-
		IMDN [13]	694K	158.8G	38.00/0.9605	33.63/0.9177	32.19/0.8996	32.17/0.9283	38.88/0.9774
		A2N-M (Ours)	**832K**	**200.3G**	**38.06/0.9601**	**33.73/0.9190**	**32.22/0.8997**	**32.34/0.9300**	**38.80/0.9765**
		A2F-M [29]	999K	224.2G	38.04/0.9607	33.67/0.9184	32.18/0.8996	32.27/0.9294	38.87/0.9774
	>1,000 K	AWSRN-M [26]	1063K	244.1G	38.04/0.9605	33.66/0.9181	32.21/0.9000	32.23/0.9294	38.66/0.9772
		SRMDNF [35]	1513K	347.7G	37.79/0.9600	33.32/0.9150	32.05/0.8980	31.33/0.9200	-
		CARN [2]	1592K	222.8G	37.76/0.9590	33.52/0.9166	32.09/0.8978	31.92/0.9256	-
		DRCN [15]	1774K	17974G	37.63/0.9588	33.04/0.9118	31.85/0.8942	30.75/0.9133	37.63/0.9723
		A2N (Ours)	**1036K**	**247.5G**	**38.06/0.9608**	**33.75/0.9194**	**32.22/0.9002**	**32.43/0.9311**	**38.87/0.9769**
3	<1,000 K	SRCNN [7]	57K	52.7G	32.75/0.9090	29.28/0.8209	28.41/0.7863	26.24/0.7989	30.59/0.9107
		FSRCANN [8]	12K	6G	33.16/0.9140	29.43/0.8242	28.53/0.7910	26.43/0.8080	30.98/0.9212
		DRRN [24]	297K	6797G	34.03/0.9244	29.96/0.8349	28.95/0.8004	27.53/0.8378	32.74/0.9390
		VDSR [14]	665K	612.6G	33.66/0.9213	29.77/0.8314	28.82/0.7976	27.14/0.8279	32.01/0.9310
		MemNet [25]	677K	2662.4G	34.09/0.9248	30.00/0.8350	28.96/0.8001	27.56/0.8376	-
		IMDN [13]	703K	71.5G	34.36/0.9270	30.32/0.8417	29.09/0.8046	28.17/0.8519	33.61/0.9445
		A2N-M (Ours)	**832K**	**96.6G**	**34.50/0.9279**	**30.41/0.8438**	**29.13/0.8058**	**28.35/0.8563**	**33.79/0.9458**
	>1,000 K	A2F-M [29]	1003K	100G	34.50/0.9278	30.39/0.8427	29.11/0.8054	28.28/0.8546	33.66/0.9453
		AWSRN-M [26]	1143K	116.6G	34.42/0.9275	30.32/0.8419	29.13/0.8059	28.26/0.8545	33.64/0.9450
		SRMDNF [35]	1530K	156.3G	34.12/0.9250	30.04/0.8370	28.97/0.8030	27.57/0.8400	-
		CARN [2]	1592K	118.8G	34.29/0.9255	30.29/0.8407	29.06/0.8034	28.06/0.8493	-
		DRCN [15]	1774K	17974G	33.82/0.9226	29.76/0.8311	28.80/0.7963	27.15/0.8276	32.31/0.9328
		A2N (Ours)	**1036K**	**117.5G**	**34.47/0.9279**	**30.44/0.8437**	**29.14/0.8059**	**28.41/0.8570**	**33.78/0.9458**
4	<1,000 K	SRCNN [7]	57K	52.7G	30.48/0.8628	27.49/0.7503	26.90/0.7101	24.52/0.7221	27.66/0.8505
		FSRCANN [8]	12K	4.6G	30.71/0.8657	27.59/0.7535	26.98/0.7150	24.62/0.7280	27.90/0.8517
		DRRN [24]	297K	6797G	31.68/0.8888	28.21/0.7720	27.38/0.7284	25.44/0.7638	29.46/0.8960
		VDSR [14]	665K	612.6G	31.35/0.8838	28.01/0.7674	27.29/0.7251	25.18/0.7524	28.83/0.8809
		MemNet [25]	677K	2662.4G	31.74/0.8893	28.26/0.7723	27.40/0.7281	25.50/0.7630	-
		IMDN [13]	715K	40.9G	32.21/0.8948	28.58/0.7811	27.56/0.7353	26.04/0.7838	30.45/0.9075
		A2N-M (Ours)	**843K**	**60.6G**	**32.27/0.8963**	**29.69/0.7842**	**27.61/0.7376**	**26.28/0.7919**	**30.59/0.9103**
	>1,000 K	A2F-M [29]	1010K	56.7G	32.28/0.8955	28.62/0.7828	27.58/0.7364	26.17/0.7892	30.57/0.9100
		AWSRN-M [26]	1254K	72G	32.21/0.8954	28.65/0.7832	27.60/0.7368	26.15/0.7884	30.56/0.9093
		SRMDNF [35]	1555K	89.3G	31.96/0.8930	28.35/0.7770	27.49/0.7340	25.68/0.7730	-
		CARN [2]	1592K	90.9G	32.13/0.8937	28.60/0.7806	27.58/0.7349	26.07/0.7837	-
		DRCN [15]	1774K	17974G	31.53/0.8854	28.02/0.7670	27.23/0.7233	25.14/0.7510	28.98/0.8816
		A2N (Ours)	**1047K**	**72.4G**	**32.30/0.8966**	**28.71/0.7842**	**27.61/0.7374**	**26.27/0.7920**	**30.67/0.9110**

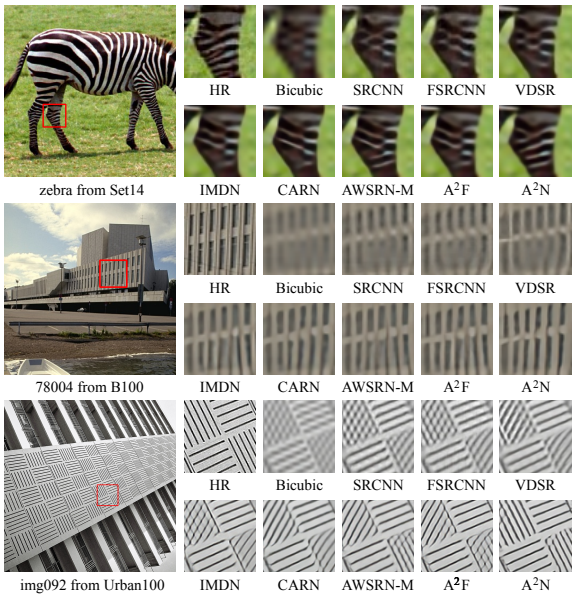

Figure 6: Visual comparison for upscaling factor $\times 4$.

feature fusion method than other methods.

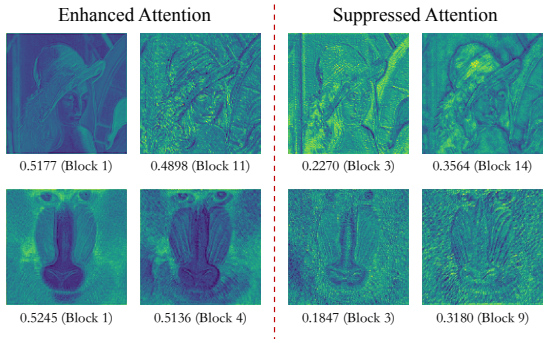

Figure 7: **Left:** the most enhanced attention maps. **Right:** the most suppressed attention maps. The weight value π^{att} and the block index are shown under the attention maps.

Results of Local Attribution Maps. The proposed method can make good use of the information in the input LR image. Recently, Gu *et al*. [9] propose a novel attribution approach called local attribution map (LAM), which performs attribution analysis of SR networks and aims at finding the input pixels that strongly influence the SR results. The results are summarised to diffusion index (DI), an evaluation metric that measures the ability of extraction

Table 6: Results of different methods used in non-attention branch. A^2N-attn-only: attention branch only. A^2N-no-op: pass through in non-attention branch. Test on Set14 ($\times$ 4).

Method	Parameter	PSNR
A^2N-attn-only	810K	28.646
A^2N-no-op (No operations)	817K	28.660
A^2N-M (1 $\times$ 1 Conv)	843K	28.695
A^2N (3 $\times$ 3 Conv)	1047K	28.707

Table 7: Test on 150 images which are proposed from LAM [9]. The DI reflects the range of involved pixels. A higher DI represents a wider range of attention.

Model	DI	PSNR
FSRCNN	0.797	20.30
CARN	1.807	21.27
IMDN	14.643	21.23
A^2F	12.58	21.43
A^2N-attn-only	2.54	21.33
A^2N-non-attn-only	1.56	20.99
A^2N-Addition	2.75	21.37
A^2N-Concatenation	2.70	21.38
A^2N-AdaptiveWeights	2.43	21.33
A^2N	14.77	21.44

and utilization of the information in the LR image. A larger DI indicates more pixels are involved. LAM highlights the pixels which have the greatest impact on the SR results. For the same local patch, if the LAM map involves more pixels or a larger range, it can be considered that the SR network has extracted and used the information from more pixels. We follow the suggested setting, and in Table 7 we show the PSNR and DI performances for some SR networks test on these images. Among all the models, A^2N has the highest DI and PSNR. We can notice that using attention dropout module makes DI much higher than other models. Figure 8 shows the LAM results, which visualize the importance of pixels. The LAM results indicate that CARN and channel-spatial attention model only utilize very limited information, Our A^2N can utilize a wider range of information for better SR result for models without downsampling.

5.5. Ablation Study

Choice of channel width in Non-attention Branch. In A^2N, we use 3 $\times$ 3 convolution to extract features, in A^2N-M, 1 $\times$ 1 convolution is used. We compare the results of 1 $\times$ 1 convolution, 3 $\times$ 3 convolution and pass-through. From our experimental results shown in Table 6, even without any operations, the A^2 structure is still better than one-trunk attention structure. Compare with one-trunk structure and 1 $\times$ 1 convolution, 1 $\times$ 1 convolution gain 0.049 dB improvement with only 33K parameters. It achieves a great trade-off between parameter number and performance. The results of w/o operation and 1 $\times$ 1 convolution also prove that convolution in non-attention branch truly contributes to the network, it can extract the effective features that the attention branch cannot extract, so that it can complement the features of attention branch. These comparisons firmly demonstrate the effectiveness of the A^2 structure.

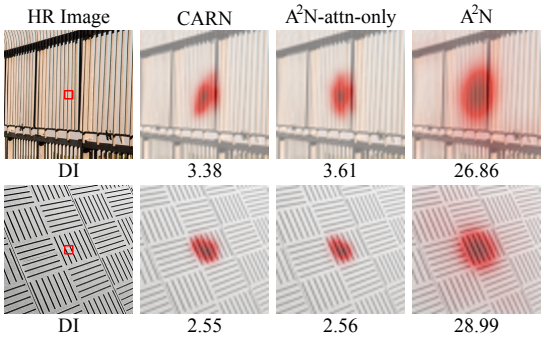

Figure 8: Results of the LAM for SR network interpretation. The LAM maps represent the importance of each pixel in the input LR image w.r.t. the SR of the patch marked with a red box. We illustrate the area of contribution in red color.

Model Size. We carry out experiments on there different sizes of models to analyse the effectiveness of our A^2 structure. We chose the model which only keeps attention branch to compare with our A^2N. We change the number of channels and A^2B and get three sizes of models: large, medium and small. From our experimental results shown in Table 8, for all models of different sizes, the A^2 structure could conspicuously improve their performance. Smaller models with the A^2 structure will gain more bonus than larger models. No matter what the size of the model, our A^2 structure only takes up a small number of parameters, so our method can be applied to various sizes of models.

Table 8: Results of base model and our A^2 structure on different model size. The base model denotes models with only one attention trunk. Test on Set14 ($\times$ 4).

Size	#channel	#block	Model	PSNR	Gain
S	40	16	base	28.646	-
			A^2N	28.707	+0.061
M	64	32	base	28.728	-
			A^2N	28.769	+0.041
L	64	64	base	28.782	-
			A^2N	28.815	+0.033

6. Conclusions

In this work, we propose attention in attention networks (A^2N) and building block A^2B for image SR. Our A^2B dynamically adjust the contribution of attention layers, allow them to be penalized less frequently. It allows to more aggressive pixel-wise attention adjustments and less chance of performance degradation without using significant amount of extra parameters. Experiments have demonstrated that our method could achieve superior performances comparing with state-of-the-art SR models of similar sizes.

References

[1] Eirikur Agustsson and Radu Timofte. Ntire 2017 challenge on single image super-resolution: Dataset and study. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, July 2017. 5

[2] Namhyuk Ahn, Byungkon Kang, and Kyung-Ah Sohn. Fast, accurate, and lightweight super-resolution with cascading residual network. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 252–268, 2018. 2, 6, 7

[3] Marco Bevilacqua, Aline Roumy, Christine Guillemot, and Marie Line Alberi-Morel. Low-complexity single-image super-resolution based on nonnegative neighbor embedding. 2012. 5

[4] Long Chen, Hanwang Zhang, Jun Xiao, Liqiang Nie, Jian Shao, Wei Liu, and Tat-Seng Chua. Sca-cnn: Spatial and channel-wise attention in convolutional networks for image captioning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5659–5667, 2017. 2

[5] Yinpeng Chen, Xiyang Dai, Mengchen Liu, Dongdong Chen, Lu Yuan, and Zicheng Liu. Dynamic convolution: Attention over convolution kernels. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11030–11039, 2020. 4, 5

[6] Tao Dai, Jianrui Cai, Yongbing Zhang, Shu-Tao Xia, and Lei Zhang. Second-order attention network for single image super-resolution. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 11065–11074, 2019. 2

[7] Chao Dong, Chen Change Loy, Kaiming He, and Xiaoou Tang. Image super-resolution using deep convolutional networks. *IEEE transactions on pattern analysis and machine intelligence*, 38(2):295–307, 2015. 2, 6, 7

[8] Chao Dong, Chen Change Loy, and Xiaoou Tang. Accelerating the super-resolution convolutional neural network. In *European conference on computer vision*, pages 391–407. Springer, 2016. 6, 7

[9] Jinjin Gu and Chao Dong. Interpreting super-resolution networks with local attribution maps. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021. 1, 2, 7, 8

[10] Jie Hu, Li Shen, and Gang Sun. Squeeze-and-excitation networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7132–7141, 2018. 2

[11] Yanting Hu, Jie Li, Yuanfei Huang, and Xinbo Gao. Channel-wise and spatial feature modulation network for single image super-resolution. *IEEE Transactions on Circuits and Systems for Video Technology*, 30(11):3911–3927, 2019. 1, 2, 3

[12] Jia-Bin Huang, Abhishek Singh, and Narendra Ahuja. Single image super-resolution from transformed self-exemplars. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5197–5206, 2015. 5

[13] Zheng Hui, Xinbo Gao, Yunchu Yang, and Xiumei Wang. Lightweight image super-resolution with information multi-distillation network. In *Proceedings of the 27th ACM International Conference on Multimedia*, pages 2024–2032, 2019. 6, 7

[14] Jiwon Kim, Jung Kwon Lee, and Kyoung Mu Lee. Accurate image super-resolution using very deep convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1646–1654, 2016. 2, 6, 7

[15] Jiwon Kim, Jung Kwon Lee, and Kyoung Mu Lee. Deeply-recursive convolutional network for image super-resolution. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1637–1645, 2016. 2, 6, 7

[16] Jun-Hyuk Kim, Jun-Ho Choi, Manri Cheon, and Jong-Seok Lee. Ram: Residual attention module for single image super-resolution. *arXiv preprint arXiv:1811.12043*, 2, 2018. 1, 6

[17] Bee Lim, Sanghyun Son, Heewon Kim, Seungjun Nah, and Kyoung Mu Lee. Enhanced deep residual networks for single image super-resolution. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 136–144, 2017. 2, 4

[18] Ding Liu, Bihan Wen, Yuchen Fan, Chen Change Loy, and Thomas S Huang. Non-local recurrent network for image restoration. In *Advances in Neural Information Processing Systems*, pages 1673–1682, 2018. 2

[19] Jie Liu, Wenjie Zhang, Yuting Tang, Jie Tang, and Gangshan Wu. Residual feature aggregation network for image super-resolution. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2359–2368, 2020. 2

[20] Hailong Ma, Xiangxiang Chu, and Bo Zhang. Accurate and efficient single image super-resolution with matrix channel attention network. In *Proceedings of the Asian Conference on Computer Vision*, 2020. 1

[21] David Martin, Charless Fowlkes, Doron Tal, and Jitendra Malik. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In *Proceedings Eighth IEEE International Conference on Computer Vision. ICCV 2001*, volume 2, pages 416–423. IEEE, 2001. 5

[22] Yusuke Matsui, Kota Ito, Yuji Aramaki, Azuma Fujimoto, Toru Ogawa, Toshihiko Yamasaki, and Kiyoharu Aizawa. Sketch-based manga retrieval using manga109 dataset. *Multimedia Tools and Applications*, 76(20):21811–21838, 2017. 5

[23] Jongchan Park, Sanghyun Woo, Joon-Young Lee, and In So Kweon. Bam: Bottleneck attention module. *arXiv preprint arXiv:1807.06514*, 2018. 2

[24] Ying Tai, Jian Yang, and Xiaoming Liu. Image super-resolution via deep recursive residual network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3147–3155, 2017. 6, 7

[25] Ying Tai, Jian Yang, Xiaoming Liu, and Chunyan Xu. Memnet: A persistent memory network for image restoration. In *Proceedings of the IEEE international conference on computer vision*, pages 4539–4547, 2017. 2, 6, 7

[26] Chaofeng Wang, Zheng Li, and Jun Shi. Lightweight image super-resolution with adaptive weighted learning network. *arXiv preprint arXiv:1904.02358*, 2019. 6, 7

[27] Fei Wang, Mengqing Jiang, Chen Qian, Shuo Yang, Cheng Li, Honggang Zhang, Xiaogang Wang, and Xiaoou Tang. Residual attention network for image classification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3156–3164, 2017. 2

[28] Xiaolong Wang, Ross Girshick, Abhinav Gupta, and Kaiming He. Non-local neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7794–7803, 2018. 2

[29] Xuehui Wang, Qing Wang, Yuzhi Zhao, Junchi Yan, Lei Fan, and Long Chen. Lightweight single-image super-resolution network with attentive auxiliary feature learning. In *Proceedings of the Asian Conference on Computer Vision*, 2020. 1, 6, 7

[30] Xintao Wang, Ke Yu, Shixiang Wu, Jinjin Gu, Yihao Liu, Chao Dong, Yu Qiao, and Chen Change Loy. Esrgan: Enhanced super-resolution generative adversarial networks. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 0–0, 2018. 2

[31] Zhihao Wang, Jian Chen, and Steven CH Hoi. Deep learning for image super-resolution: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2020. 2

[32] Sanghyun Woo, Jongchan Park, Joon-Young Lee, and In So Kweon. Cbam: Convolutional block attention module. In *Proceedings of the European conference on computer vision (ECCV)*, pages 3–19, 2018. 2

[33] Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhudinov, Rich Zemel, and Yoshua Bengio. Show, attend and tell: Neural image caption generation with visual attention. In *International conference on machine learning*, pages 2048–2057. PMLR, 2015. 2

[34] Jianchao Yang, John Wright, Thomas S Huang, and Yi Ma. Image super-resolution via sparse representation. *IEEE transactions on image processing*, 19(11):2861–2873, 2010. 5

[35] Kai Zhang, Wangmeng Zuo, and Lei Zhang. Learning a single convolutional super-resolution network for multiple degradations. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3262–3271, 2018. 6, 7

[36] Yulun Zhang, Kunpeng Li, Kai Li, Lichen Wang, Bineng Zhong, and Yun Fu. Image super-resolution using very deep residual channel attention networks. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 286–301, 2018. 1, 2, 3

[37] Y Zhang, K Li, K Li, B Zhong, and Y Fu. Residual non-local attention networks for image restoration. In *International Conference on Learning Representations*, 2019. 2

[38] Yulun Zhang, Yapeng Tian, Yu Kong, Bineng Zhong, and Yun Fu. Residual dense network for image super-resolution. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2472–2481, 2018. 2

[39] Hengyuan Zhao, Xiangtao Kong, Jingwen He, Yu Qiao, and Chao Dong. Efficient image super-resolution using pixel attention. *arXiv preprint arXiv:2010.01073*, 2020. 2, 3, 6