IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received August 21, 2019, accepted September 14, 2019, date of publication September 19, 2019, date of current version October 2, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2942346

HRAN: Hybrid Residual Attention Network for Single Image Super-Resolution

ABDUL MUQEET, MD TAUHID BIN IQBAL, AND SUNG-HO BAE
Department of Computer Science and Engineering, Kyung Hee University, Yongin 17104, South Korea

Corresponding author: Sung-Ho Bae (shbae@khu.ac.kr)

This work was supported in part by the Basic Science Research Program under the National Research Foundation of Korea (NRF) through the Ministry of Science, ICT and Future Planning under Grant 2018R1C1B3008159.

ABSTRACT The extraction and proper utilization of convolutional neural network (CNN) features have a significant impact on the performance of image super-resolution (SR). Although CNN features contain both spatial and channel information, current deep learning techniques for SR often suffer to maximize the performance due to using either the spatial information or channel information. Moreover, they integrate such information within a deep or wide network rather than exploiting all the available features, eventually resulting in high computational complexity. To address these issues, we present a binarized feature fusion (BFF) structure that utilizes the extracted features from global residuals (GR) in an effective way. Each GR consists of multiple hybrid residual attention blocks (HRAB) that effectively integrates the multiscale feature extraction module and channel attention mechanism in a single block. Furthermore, to save computational power, instead of using a large filter size, we use convolutions with different dilation factors to extract multiscale features. We also propose to adopt global skip connections (GSC), short skip connections (SSC), long skip connections (LSC) and GR structure to ease the flow of information without losing important features details. In the paper, we call this overall network architecture as hybrid residual attention network (HRAN). In the experiment, we have observed the efficacy of our method against the state-of-the-art methods for both the quantitative and qualitative comparisons.

INDEX TERMS Single image super-resolution, spatial attention, channel attention, hybrid attention, feature fusion, efficient feature extraction.

I. INTRODUCTION

In this paper, we address the Single Image Super-Resolution (SISR) problem, where the objective is to reconstruct the accurate high-resolution (HR) image from a single low-resolution (LR) image. It is known as an ill-posed problem, since there are multiple solutions available for mapping any LR image to HR images. This problem is intensified when the up-sampling factor becomes larger. Because HR images preserve much richer information than LR images, SISR techniques are popular in many practical applications, such as surveillance [38], Face Hallucination [31], Hyperspectral imaging [13], medical imaging [23] etc.

Numerous deep learning based methods have been proposed in recent years to address the SISR problem. Among them, SRCNN [3] is considered as the first attempt to come up with a deep-learning based solution with its three convolutional layers. SRCNN outperformed the existing SISR approaches that typically used either multiple images with different scaling factors and/or handcrafted features. Later, Dong *et al.* [4] proposed an architecture named VDSR that extended the depth of CNN up to twenty layers while adding a global residual connection within the architecture. DRCN [11] also increased the depth of network through a recursive supervision and skip connection, and improved the performance. However, due to increasing depth of the networks, vanishing gradient resisted the network to be converged [7]. In the image classification domain, to solve the aforementioned problem, He *et al.* [7] proposed a residual block by which a network over 1000 layers was successfully trained. Inspired by its very deep architecture with residual blocks, EDSR [16] proposed much wider and deeper networks for the SISR problem using residual blocks, called EDSR and MDSR [16], respectively.

The associate editor coordinating the review of this manuscript and approving it for publication was Pasquale De Meo.

Very recently, Zhang *et al.* [35] proposed RCAN that utilizes a channel attention block to exploit the inter-dependencies across the feature channels. Moreover, Li *et al.* [15] proposed MSRN that improved the reconstruction performance by exploiting the information of spatial features rather than increasing the depth of CNNs. MSRN combined the features extracted from different convolution filter sizes and concatenates the outputs of all residual blocks through a hierarchical feature fusion (HFF) technique, utilizing the information of the intermediate feature maps. By doing so, MSRN achieved comparable performance against EDSR [16] although having a 7-times smaller model size. In [37], Zhang *et al.* proposed DCSR in which they proposed a mixed convolution block that combines dilated and conventional convolutional layers to attain larger receptive field sizes. Nonetheless, most of these CNN-based methods focused either on increasing the number of layers [10], [11], [16], [35] or on extending the width and height in a layer of CNN to achieve higher performance [15]. In this way, they put less focus on exploiting the by-product CNN features, i.e. spatial and channel information simultaneously, and thus suffer to maximize the performance at times.

Moreover, the strong correlations between the input LR and output HR images [15] lead us to assuming that, apart from the high-level features, low-level and mid-level features also play vital roles for reconstructing a super-resoled image. Therefore, we argue that the mid- and high-level features should also be treated precisely in this paper.

In the previous work, dense connections were used [28], which added every feature to subsequent features with residual connections. As a variant of dense connections, HFF [15], [28], [36] was proposed to remove the trivial residual connections and to directly concatenate all the output features from the residual blocks for the SISR problem. However, this direct feature concatenation prohibits the features from smooth feature transformation from low to high levels, resulting in improper utilization of various low-level and mid-level features. This may introduce redundancy in feature utilization, thus increasing the cost of computation complexity. In our ablation study in Section 4.1, this problem will be analyzed in details.

To solve this problem, in this paper, we propose a BFF structure that combines adjacent feature maps with 1×1 convolutions, which is repeatedly performed until remaining a single feature map. This allows all the features extracted from CNN to be integrated smoothly, thus fully utilizing various features with different levels. Moreover, to efficiently extract the features, unlike previous work that used only residual blocks in [15] as feature extractors, we come up with GR that are constructed with the proposed hybrid residual attention block (HRAB). Our proposed HRAB extracts both spatial and channel information with the notion that the both information is important in the reconstruction of high quality SR images and should be extracted simultaneously in a single module.

Moreover, compared to MSRN [15] that concatenates the conventional convolutional layers with different kernel sizes to enlarge receptive field sizes, our proposed method concatenates convolutional layers with different dilation factors exploiting much larger receptive fields while significantly decreasing the number of convolution weights. Furthermore, to ease the flow of information, we introduce the SSC, LSC, and GSC skip connections. We conduct comprehensive experiments to verify the efficacy of our method, where we observe its superiority against other state-of-the-art methods.

We summarize the overall contributions of this work as,

- We propose a BFF to transfer all the low- mid- and high-level features to the end of the network. This allows the network to smoothly transform the features with different levels and generate an effective feature map in the final reconstruction stage.
- The proposed HRAB considers both channel and spatial attention mechanisms to exploit the channel and spatial dependencies. The spatial attention mechanism extracts the fine spatial features with larger receptive field sizes whereas the channel attention guides in selecting the most important feature channels thus in the end, we have more discriminative features.
- Unlike the previous work [15] which utilizes the features of residual blocks for HFF, we employ BFF on GRs to avoid the redundant information, resulting in better reconstruction performance.
- For extracting the multiscale spatial features, we propose to use a mixed dilated convolution block with different dilation factors. Compared to the previous work in [15] that used the large kernel sizes to secure large receptive fields, our proposed method can achieve similar performance even with smaller kernel sizes. Moreover, we propose to use the dilated convolution in an effective manner to avoid the gridding problem of the conventional dilated convolutional layers.
- To ease the transmission of information throughout the network, we propose to adopt the hierarchical (GSC, SSC and LSC) skip connections in our architecture.

II. RELATED WORK

Several CNN-based SISR methods have been proposed in the recent past. In 2014, Dong *et al.* [3] proposed SRCNN, the first CNN network architecture in the SR domain. It was a shallow 3 layers CNN architecture which achieved the superior performance against the previous non-CNN methods. Later, based on a residual learning technique in [7], Kim *et al.* [10], [11] achieved remarkable performance with their proposed VDSR and DRCN methods. VDSR used a deeper (20 layers) CNN and global residual connection method whereas DRCN [11] used a recursive block to increase the depth of CNN. Thanks to the recursive block, DRCN does not require many convolution weights to secure large repetitive fields. Tai *et al.* [24] proposed the MemNet which had memory blocks that consist of recursive and gate

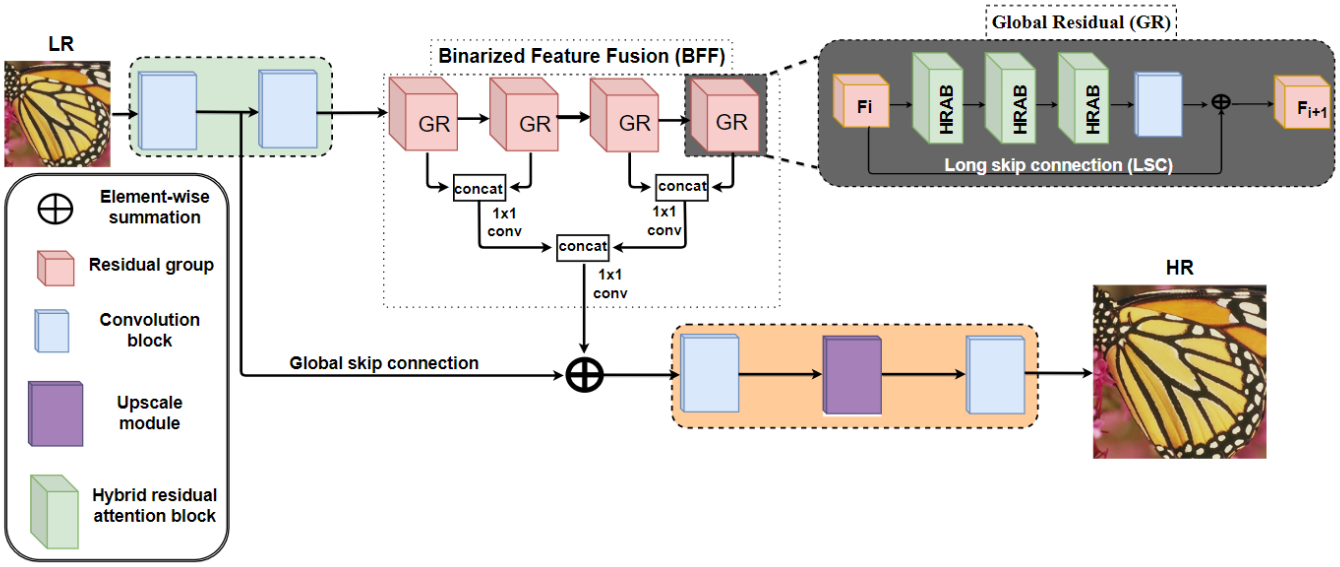

FIGURE 1. The proposed network architecture HRAN. The green-shaded area at top-left performs shallow feature extraction. the gray-shaded area at top-right indicates the internal structure of GR. The proposed BFF smoothly integrates features from low to high level GR blocks, and the output of BFF is element-wise summed with the shallow features and is fed into the final reconstruction stage (the orange-shaded area) to produce an HR image. The left-bottom block shows the specific descriptions.

units. All of these methods have used the interpolated LR image as input to meet the size of the target HR image. Due to this preprocessing, these methods flow high dimensional tensors in the network, introducing additional computation complexity with some visual artifacts [22].

On the other hand, the recent state-of-the-art methods directly learn the mapping from the LR image input without upscaling. Dong *et al.* [4] proposed FSRCNN, an improved version of SRCNN, having faster training and inference time by allowing the network to have an LR input image without upscaling. Ledig *et al.* [14] proposed SRResNet, inspired from ResNet [7], to construct the deeper network. With the perceptual loss function in Generative Adversarial Networks (GAN), they proposed SGRAN for photo-realistic SR. Lim *et al.* [16] removed the trivial modules (like batch normalization) of SRResNet, and proposed EDSR (wider) and MDSR (deeper) that made a significant improvement in the SR problem. EDSR has a large number of filters (256 filters) whereas MDSR has a small number of filters though the depth of CNN network is increased to 165 layers. It has shown that deeper networks can achieve remarkable performance. Consequently, Zhang *et al.* [35] proposed a very deep network for SR. To the extent of our knowledge, it has the largest depth in the SR domain. Zhang *et al.* [35] has shown that only stacking the layers cannot improve the performance and proposed to use the CA [8] mechanism to neglect the low-frequency information while selecting the valuable high-frequency feature maps. To increase the depth of the network, they proposed the residual in residual (RIR) structure. Nevertheless, their network, called RCAN [35], is very deep, thus making it difficult to use it in real-life applications due to very slow inference time.

In contrast, multiscale feature extraction techniques, which are less explored in SISR, have shown significant performance in object detection [17], image segmentation [21], and model compression [2] to achieve good tradeoffs between speed and accuracy. Li et al. proposed a multiscale residual network (MSRN) [15] having only 8 residual blocks. MSRN used multipath convolutional layers with different kernel sizes (3×3 and 5×5) to extract the multiscale spatial features. Furthermore, it proposed to use the HFF architecture to utilize the intermediate features. The intuition behind HFF architecture is to transfer the middle features at the end of the network since the increase in the depth of the network may cause vanishing intermediate features. HFF shows comparable performance to EDSR. However, as the depth or width of a network increases, HFF also increases the computation complexity.

In this paper, we found that it is also important to build an efficient SR network to fully utilize the feature information as well as channel information. Considering it, we propose a hybrid residual attention network (HRAN) which combines the multiscale feature extraction along with the channel attention [8] mechanism. In this paper, we refer the multiscale feature extraction as spatial attention and call the combination of the channel and spatial attention, hybrid attention.

III. HYBRID RESIDUAL ATTENTION NETWORK

A. NETWORK ARCHITECTURE

The proposed HRAN architecture is shown in Figure 1. The HRAN can be decomposed into two parts: feature extraction and reconstruction. The feature extraction is further divided into two parts: shallow feature extraction and deep feature extraction. The deep feature extraction includes GR with

BFF structure. Whereas, GR contains a sequence of HRAB followed by 3×3 convolution. We represent the input and output of HRAN as I_{LR} and I_{SR} respectively. We aim to reconstruct the accurate HR image I_{HR} directly from LR image I_{LR} without upscaling.

In the shallow feature extraction, we use two convolutional layers to extract the features from input I_{LR} image.

$$F_0 = H_{SF1}\left(I_{LR}\right), \tag{1}$$

Here $H_{SF1}(\cdot)$ represents the convolution operation. The output feature F_0 in Eq. (1) is used for global residual learning to preserve the input features. As mentioned above, we pass F_0 for further feature extraction as

$$F_1 = H_{SF2}\left(F_0\right), \tag{2}$$

where $H_{SF2}(\cdot)$ represents the convolution operation. F_1 is the output of the shallow feature extraction step and will be used as input for the deep feature extraction as

$$F_{DF} = H_{DF}\left(F_1\right) + F_0, \tag{3}$$

where $H_{DF}(\cdot)$ represents the deep feature extraction function and F_0 shows global residual connection to the end of deep feature as in VDSR [10]. The deep features are sequentially extracted through HRAB, GR and BFF, which will be fully described in the subsequent sections. Consequently, the last reconstruction step can be expressed as

$$I_{SR} = H_{REC}\left(F_{DF}\right), \tag{4}$$

where H_{REC} denotes the reconstruction function. We reconstruct the I_{SR} to have the same dimension with the I_{HR} through deep features of I_{LR}. There are various techniques to serve as upsampling modules, such as PixelShuffle layer [22], deconvolutional layer [4], nearest-neighbor upsampling convolutional [5]. In this work, we use MSRN [15] reconstruction module that enables us to upscale to any upscale factor with minor changes. The proposed HRAN function can be expressed as

$$I_{SR} = H_{HRAN}\left(I_{LR}\right), \tag{5}$$

For the optimization, numerous loss functions have been discussed for SISR. The mostly used loss functions are L_1, and L_2 distance norms whereas perceptual and adversarial losses are also preferred. To keep the network simple and avoid the trivial training tricks, we adopt the L_1 loss function for training HRAN. L_1 loss function is defined as

$$L_1(\Theta) = \frac{1}{N}\sum_{i=1}^{N}\left\|H_{HRAN}\left(I_{LR}^{i}\right) - I_{HR}^{i}\right\|_1, \tag{6}$$

where Θ denotes the weights and bias of our network, and N is the total number of image patches used for training HRAN.

B. BINARIZED FEATURE FUSION (BFF) STRUCTURE

The shallow features lack the fine details for SISR. We use deep networks to solve this problem. Since there is a strong correlation between I_{LR} and I_{SR} in SISR, it is required to fully utilize the features of I_{LR} and transmit them to the end of the network. However, due to its deep depth in network, the low-level features are lost during the transmission, results in inaccurate reconstruction. One possible solution is to use a residual connection, however, it induces the redundant information [15]. The MSRN [15] uses HFF to transmit the information from all the feature maps towards the end of the network. However, the concatenation of every feature generates a lot of redundant information and also increase memory computation.

To ease this problem, we propose the BFF structure as shown in Figure 1. First, we concatenate the adjacent GR blocks and then, we remove the redundant information from adjacent blocks using 1×1 convolution. We repeat this procedure for all GR blocks and the resultant blocks produced through this mechanism until all the blocks are integrated into the single GR block, which is convolved by 1×1 to produce the output features. In the end, we element-wise add this output to the shallow features' output (F_0). We refer this element-wise summation as GSC in Figure 1. The notable difference in the proposed BFF structure and previous ones [15] is the use of GR instead of multi-scale residual block (MSRB) in MSRN. The use of GRs does not only help to increase the depth of the network without the vanishing gradient problem but also reduce the memory overhead when concatenating the features map.

The first step of BFF is the feature extraction through the GR block. We explain the details of GR in the next section. When we extract all the features through GR blocks, we can utilize these GR blocks with BFF architecture. The proposed BFF can be represented as

$$M_j = H_{1\times1}\left[F_{i+1}, F_{i+2}\right], \tag{7}$$

$$M_{j+1} = H_{1\times1}\left[F_{i+3}, F_{i+4}\right], \tag{8}$$

Here, F_{i+1}, F_{i+2} are the adjacent feature maps generated through GR. The output of the two adjacent GR blocks are channel-wise concatenated and then passed into a 1 × 1 convolutional layer to avoid the redundant information. Thus, we use four GR blocks that produce two more blocks which are then processed in a similar manner such that F_{i+1} becomes M_j and F_{i+2} becomes M_{j+1} in the next step as shown in Figure 1. Thus, in the next step, M_j and M_{j+1} will act as two GR blocks. We repeat this procedure until all GRs and resultant blocks are integrated into a single output which is further used as the input of the reconstruction step.

C. GLOBAL RESIDUALS (GR)

It is shown in [16] that the stacked residual blocks enhance the performance of SR but after some extent, cause crucial information loss during transmission of features and also makes the training slower, affecting the performance gain in

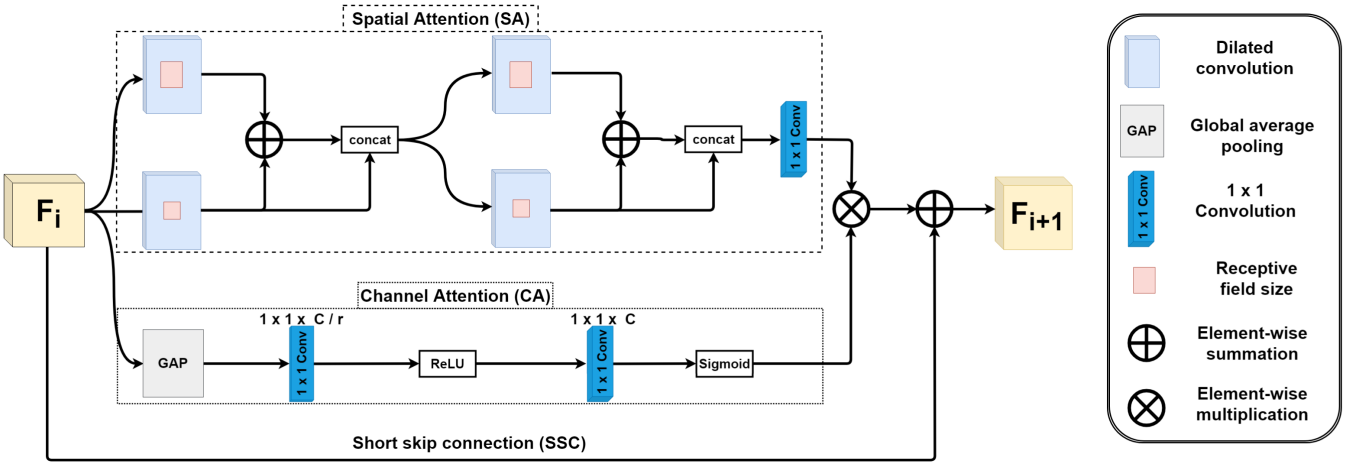

FIGURE 2. Proposed multi-path hybrid residual attention block (HRAB). Top path represents Spatial Attention (SA) that contains dilated convolutions with different dilation factors. Bottom path represents Channel Attention (CA) mechanism. Notations about different components are given in the right.

the SISR [35]. Thus, rather than increasing the depth, we propose the GR (see shaded area of Figure 1) in our architecture to detect deep features. The GR consists of multiple HRAB that are followed by 1×1 convolution. We found that only cascading many HRAB does degrade the SR performance. Thus, to preserve the information, we apply element-wise summation between the input of GR and output of 1×1 convolutional and refer it as LSC as shown in Figure 1.

The GR enables the network to remember the information through LSC whereas to detect deep features, it uses SSC within its modules, in this case, HRAB. Hence, the flow of information in GR is smoothly carried out through LSC and SSC. The details of the HRAB are discussed in the next section.

We express the single GR block as

$$H_{GR} = W_{GR} * H_n \left(H_{n-i} \left(\cdots H_i \left(F_1\right) \cdots\right)\right), \tag{9}$$

Here we have H_i represents the 'n' HRAB blocks, which takes input features from previous GR block (F_i) and produces the output (F_{i+1}). After stacking the 'B' HRAB modules, we apply 3×3 convolutions with weights W_{GR}. After applying LSC, the equation 9 can be rewritten as

$$H_{GR} = W_{GR} * H_n \left(H_{n-1} \left(\cdots H_1 \left(F_1\right) \cdots\right)\right) + F_1, \tag{10}$$

The above equation represents the first GR block because it takes the shallow features F_1 as input. Since, we have multiple GR blocks to extract the deep features, hence, the above equation can be generally written as

$$H^i_{GR} = W^i_{GR} * H^{i-1}_{GR} + H^{i-1}_{GR} \tag{11}$$

Here $i = 1, 2, \cdots, R$. We have 'R' GR blocks and each GR block uses the output of the previous block as its input except the first GR block that uses the shallow features F_1 as input. Thus, for the first GR block, $H^0_{GR} = F_1$.

D. HYBRID RESIDUAL ATTENTION BLOCK (HRAB)

In this section, we propose a multiscale multipath residual attention block for the feature extraction, called HRAB (see Figure 2). Our HRAB has two separate paths for the SA and CA mechanisms which are combined in such a way so that we could utilize most important features. Mathematically,

$$H_{HRAB}\left(F_{i+1}\right) = H_{SA}\left(F_i\right) \cdot H_{CA}\left(F_i\right) \tag{12}$$

where H_{SA} and H_{CA} denote the functions of SA and CA respectively and '$\cdot$' represents the element-wise multiplication between the SA and CA functions. Unlike RCAN [35], we propose to use element-wise multiplication between the outputs of SA and CA to extract the most informative spatial features. As shown in Figure2 we add the SSC in HRAB to ease the flow of information through the network.

1) SPATIAL ATTENTION (SA)

MSRN [15] proves that multiscale features improve the performance with lesser residual blocks. In MSRN [15], authors use the multiple CNN filters with increasing kernel sizes (3×3 and 5×5) to extract multiscale features. The intuition behind the larger kernel size is to take advantage of large receptive fields. But, the large kernel size causes to increase the memory computation. Thus, we propose to use the dilated convolutional layers with different dilation factors which can have the same receptive fields as large kernel size and memory consumption is similar to smaller kernel size. But, only stacking the dilated convolutional layers produces gridding effect [32]. To avoid this problem, as illustrated in Figure 2, we propose to use the element-wise sum operation between the dilated convolutions with different factors before the concatenation operation. If F_{i-1} and F_i are the input and output of SA respectively then SA can be described as :

$$S_1 = LeakyReLU\left(H_{DC1}\left(F_i\right)\right) \tag{13}$$

$$S_2 = LeakyReLU\left(H_{DC2}\left(F_i\right) + S_1\right) \tag{14}$$

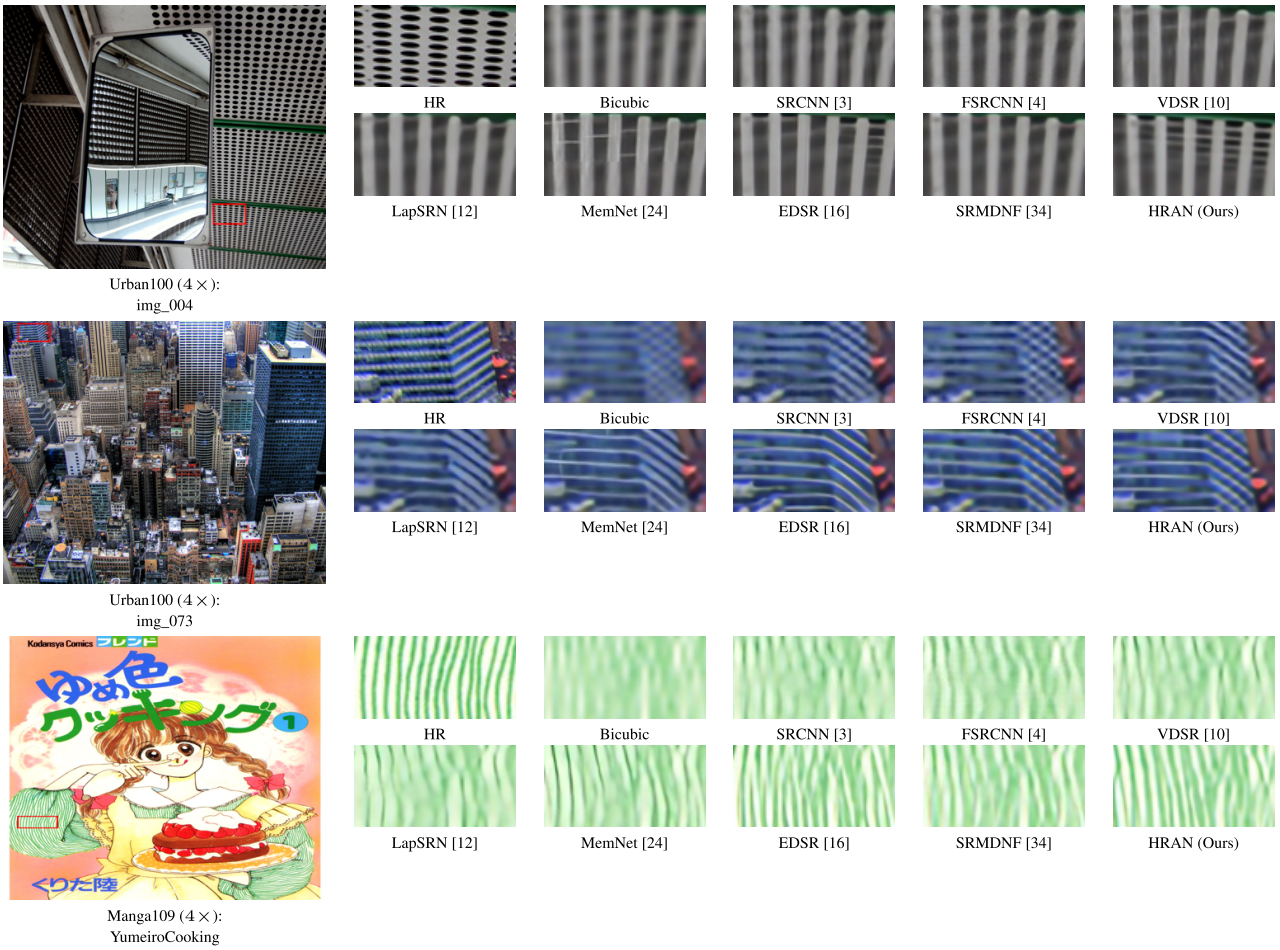

FIGURE 3. Qualitative results for 4× SR with BI model on Urban100 and Manga109 datasets.

$$S = \left[S_1, S_2\right] \tag{15}$$

$$S_1 = LeakyReLU\left(H_{DC1}\left(S\right)\right) \tag{16}$$

$$S_2 = LeakyReLU\left(H_{DC2}\left(S\right) + S_1\right) \tag{17}$$

$$H_{SA}\left(F_{i+1}\right) = H_{1\times1} * \left[S_1, S_2\right] \tag{18}$$

where H_{DC1} and H_{DC2} denotes the convolutional layers with dilation factors 1 and 2 respectively. First, we concatenate the output of two convolutional layers to increase the channel size and at the end, we use 1×1 convolution to reduce the channels. Thus, our input and output have the same number of channels. Our SA architecture inspires from [6] which has shown that upsampling and downsampling module within the architecture improves the accuracy in SR. For the activation unit, by following [12], [25], we opt the LeakyReLU over ReLU activation whereas we use the linear bottleneck layers as suggested in [20].

2) CHANNEL ATTENTION (CA)

The CA mechanism achieves a lot of success in image classification [8]. In SISR, RCAN [35] introduces the CA layer in the network. CA plays an important role in exploiting the interchannel dependencies because some of them have trivial information while others have the most valuable information. Therefore, we decide to use channel-wise features and incorporate the CA mechanism with SA module in our HRAB. By following [8], [35], we use the global pooling average to consider the channel-wise global information. We also experiment with global pooling variance as we thought global variance could extract more high-frequencies, in contrast, we get poor results as compared with global pooling average.

If we have C channels in the feature maps $[x_1, x_2, \cdots, x_C]$ then we can express each 'c' feature map as a single value.

$$z_c\left(x_c\right) = \frac{1}{H \times W} \sum_{i=1}^{H} \sum_{j=1}^{W} x_c\left(i, j\right), \tag{19}$$

(where x_c is the spatial position (i, j) of the feature maps.

To extract the channel-wise dependencies, we use the similar sigmoid gating mechanism as [8], [35]. Alike SA, here, we replace the ReLU with LeakyReLU activation.

$$H_{CA}\left(F_{i+1}\right) = f\left(W_U LR\left(W_D z\right)\right), \tag{20}$$

Here $LR\left(\cdot\right)$ and $f\left(\cdot\right)$ represent the LeakyReLU and sigmoid gating function respectively whereas W_D and W_U respectively denote the weights of downscaling and upscaling

convolutions. It is noted that it is channel-wise downscaling and upscaling with reduction ratio r.

E. IMPLEMENTATION DETAILS

For training the HRAN network, we employ 4 GR blocks in our main architecture and in each GR block, there are 8 HRAB modules which are followed by one 3×3 convolution. For the dilated convolutional layers, we use the 3×3 convolution with dilation factor 1 and 2. We have also experimented with larger dilation factor but it gives a gridding effect. We believe, we do need to linearly increase the number of dilation factors i.e. 1, 2, and 3 rather than using 1 and 3 or 2 and 3. We use $C = 64$ filters in all the layers except the final layer which has 3 filters to produce a color image though our network can work for both gray and color images. For the channel-downscaling in CA mechanism, we set a reduction factor $r = 4$.

IV. EXPERIMENTAL RESULTS

In this section, we explain the experimental analysis of our method. For this purpose, we use public datasets that are considered as the benchmark in SISR. We provide the results of both the quantitative and qualitative experiments for the comparison of our method with several state-of-the-art networks. For the datasets, we follow the recent trends [15], [16], [26], [34], [36] and use DIV2K dataset as the training set, since it contains the high-resolution images. For testing, we choose widely used standard datasets: Set5 [1], Set14 [33], BDS100 [18], Urban100 [9] and Manga109 [19]. For the degradation, we use the Bicubic Interpolation (BI).

We evaluate our results with peak signal-to-noise ratio (PSNR) and structural similarity (SSIM) [29] on luminance channel i.e. Y of transformed YCbCr space and we remove P-pixels from each border (P refers to upscaling factor). We provide the results for scaling factor $\times 2$, $\times 3$, $\times 4$, and $\times 8$.

For the training, we follow the training settings in [15]. We extract 16 *LR* patches randomly in each training batch with the size of 64×64. We use ADAM optimizer with learning rate $lr = 10^{-4}$ which decreases to half after every 2×10^5 iterations of back-propagation. We use PyTorch framework to implement our models with NVIDIA GeForce RTX 2080 Ti GPU.

A. COMPARISON WITH STATE-OF-THE-ART METHODS

We compare our method with 10 state-of-the-art SISR methods: SRCNN [3], FSRCNN [4], VDSR [10], LapSRN [12], MEMNet [24], EDSR [16], SRMDNF [34], RDN [36], DCSR [37] and MSRN [15]. By following [16], [27], we also use similar self-ensemble strategy to improve the accuracy of our model at test time and denote with HRAN+ in Table.1.

We show our quantitative evaluation results in Table 1 for the scale factor of $\times 2$, $\times 3$, $\times 4$, and $\times 8$. It is evident from the results of our efficient model that our method outperforms most of the previous methods. Our self-ensemble model achieves the highest PSNR amongst all the models whereas our single model has comparable performance

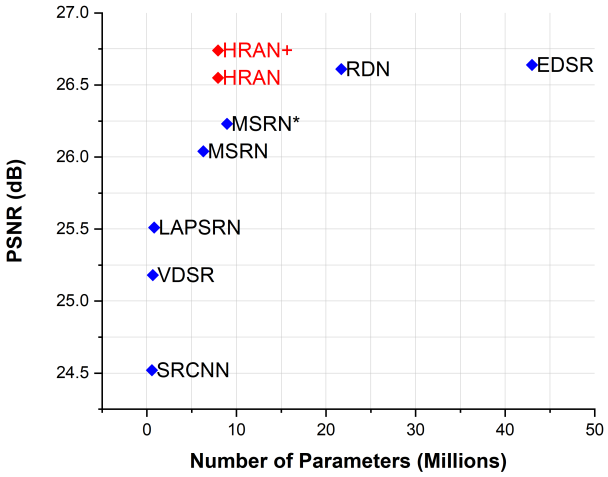

FIGURE 4. Comparison of memory and performance. Results are evaluated on Urban100 (×4).

against RDN [36]. Note that, RDN [36] has more than 22M parameters as shown in Figure 4, in contrast, our HRAN model has only 7.94 M parameters. Instead of increasing the depth and dense connections, our HRAN model with HRAB and BFF detect the deep features without increasing the depth of the network. Hence, this observation indicates that we can improve the network performance with HRAB and GR along with BFF without increasing the network depth. This also suggests that our network can further improve the accuracy with more HRAB's and GR's, though, we aim to achieve the greater accuracy by considering the memory computations.

Moreover, we present the qualitative results in Figure 3. The results of other methods are derived from [35]. In Figure 3, it can be observed from 'img_004' image our HRAN method recovers the lattices in more details, meanwhile, other methods experience the blurring artifacts. Similar behavior is also observed in 'Yumeiro-Cooking' image where other methods produce blurry lines and our HRAN produces the lines similar to HR image. It shows that our model reconstructs the fine details in output SR image through extracted deep features with GRs which are then efficiently utilized by BFF.

We note that that RCAN has shown consistently superior performance than ours in the literature [35]. However, existing methods in [39]–[41] argued that the higher performance of RCAN comes mainly from their extended number of layers and parameters, which introduce extensive computational costs making RCAN impractical. As our proposed method is positioned in efficient super resolution methods, we exclude RCAN from the comparison in this paper.

B. ABLATION STUDIES

We conduct a series of ablation studies to show the effectiveness of our model. In the first experiment, we train our model with and without CA and compare their performance with our HRAB module. For the testing, we use Urban100

TABLE 1. Quantitative comparisons of state-of-the-art methods for BI degradation model. Best, 2nd best and 3rd best results are respectively shown with Magenta, Blue, and Green colors.

Method	Scale	Set5		Set14		B100		Urban100		Manga109	
		PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
Bicubic	×2	33.66	0.9299	30.24	0.8688	29.56	0.8431	26.88	0.8403	30.80	0.9339
SRCNN [3]	×2	36.66	0.9542	32.45	0.9067	31.36	0.8879	29.50	0.8946	35.60	0.9663
FSRCNN [4]	×2	37.05	0.9560	32.66	0.9090	31.53	0.8920	29.88	0.9020	36.67	0.9710
VDSR [10]	×2	37.53	0.9590	33.05	0.9130	31.90	0.8960	30.77	0.9140	37.22	0.9750
LapSRN [12]	×2	37.52	0.9591	33.08	0.9130	31.08	0.8950	30.41	0.9101	37.27	0.9740
MemNet [24]	×2	37.78	0.9597	33.28	0.9142	32.08	0.8978	31.31	0.9195	37.72	0.9740
EDSR [16]	×2	38.11	0.9602	33.92	0.9195	32.32	0.9013	32.93	0.9351	-/-	-/-
SRMDNF [34]	×2	37.79	0.9601	33.32	0.9159	32.05	0.8985	31.33	0.9204	38.07	0.9761
RDN [36]	×2	38.24	0.9614	34.01	0.9212	32.34	0.9017	32.89	0.9353	39.18	0.9780
DCSR [37]	×2	37.54	0.9587	33.14	0.9141	31.90	0.8959	30.76	0.9142	-/-	-/-
MSRN [15]	×2	38.08	0.9605	33.74	0.9170	32.23	0.9013	32.22	0.9326	38.82	0.9868
HRAN (ours)	×2	38.21	0.9613	33.85	0.9200	32.34	0.9016	32.95	0.9357	39.12	0.9780
HRAN+ (ours)	×2	38.25	0.9614	33.99	0.9211	32.38	0.9020	33.12	0.9370	39.29	0.9785
Bicubic	×3	30.39	0.8682	27.55	0.7742	27.21	0.7385	24.46	0.7349	26.95	0.8556
SRCNN [3]	×3	32.75	0.9090	29.30	0.8215	28.41	0.7863	26.24	0.7989	30.48	0.9117
FSRCNN [4]	×3	33.18	0.9140	29.37	0.8240	28.53	0.7910	26.43	0.8080	31.10	0.9210
VDSR [10]	×3	33.67	0.9210	29.78	0.8320	28.83	0.7990	27.14	0.8290	32.01	0.9340
LapSRN [12]	×3	33.82	0.9227	29.87	0.8320	28.82	0.7980	27.07	0.8280	32.21	0.9350
MemNet [24]	×3	34.09	0.9248	30.00	0.8350	28.96	0.8001	27.56	0.8376	32.51	0.9369
EDSR [16]	×3	34.65	0.9280	30.52	0.8462	29.25	0.8093	28.80	0.8653	-/-	-/-
SRMDNF [34]	×3	34.12	0.9254	30.04	0.8382	28.97	0.8025	27.57	0.8398	33.00	0.9403
RDN [36]	×3	34.71	0.9296	30.57	0.8468	29.26	0.8093	28.80	0.8653	34.13	0.9484
DCSR [37]	×3	33.94	0.9234	30.28	0.8354	28.86	0.7985	27.24	0.8308	-/-	-/-
MSRN [15]	×3	34.38	0.9262	30.34	0.8395	29.08	0.8041	28.08	0.8554	33.44	0.9427
HRAN (ours)	×3	34.69	0.9292	30.54	0.8463	29.25	0.8089	28.76	0.8645	34.08	0.9479
HRAN+ (ours)	×3	34.75	0.9298	30.60	0.8474	29.29	0.8098	28.96	0.8670	34.36	0.9492
Bicubic	×4	28.42	0.8104	26.00	0.7027	25.96	0.6675	23.14	0.6577	24.89	0.7866
SRCNN [3]	×4	30.48	0.8628	27.50	0.7513	26.90	0.7101	24.52	0.7221	27.58	0.8555
FSRCNN [4]	×4	30.72	0.8660	27.61	0.7550	26.98	0.7150	24.62	0.7280	27.90	0.8610
VDSR [10]	×4	31.35	0.8830	28.02	0.7680	27.29	0.0726	25.18	0.7540	28.83	0.8870
LapSRN [12]	×4	31.54	0.8850	28.19	0.7720	27.32	0.7270	25.21	0.7560	29.09	0.8900
MemNet [24]	×4	31.74	0.8893	28.26	0.7723	27.40	0.7281	25.50	0.7630	29.42	0.8942
EDSR [16]	×4	32.46	0.8968	28.80	0.7876	27.71	0.7420	26.64	0.8033	-/-	-/-
SRMDNF [34]	×4	31.96	0.8925	28.35	0.7787	27.49	0.7337	25.68	0.7731	30.09	0.9024
RDN [36]	×4	32.47	0.8990	28.81	0.7871	27.72	0.7419	26.61	0.8028	31.00	0.9151
DCSR [37]	×4	31.58	0.8870	28.21	0.7715	27.32	0.7264	27.24	0.8308	-/-	-/-
MSRN [15]	×4	32.07	0.8903	28.60	0.775	27.52	0.7273	26.04	0.7896	30.17	0.9034
HRAN (ours)	×4	32.43	0.8976	28.76	0.7863	27.70	0.7407	26.55	0.8006	30.94	0.9143
HRAN+ (ours)	×4	32.56	0.8991	28.86	0.7880	27.76	0.7420	26.74	0.8046	31.26	0.9172
Bicubic	×8	24.40	0.6580	23.10	0.5660	23.67	0.5480	20.74	0.5160	21.47	0.6500
SRCNN [3]	×8	25.33	0.6900	23.76	0.5910	24.13	0.5660	21.29	0.5440	22.46	0.6950
FSRCNN [4]	×8	20.13	0.5520	19.75	0.4820	24.21	0.5680	21.32	0.5380	22.39	0.6730
SCN [30]	×8	25.59	0.7071	24.02	0.6028	24.30	0.5698	21.52	0.5571	22.68	0.6963
VDSR [10]	×8	25.93	0.7240	24.26	0.6140	24.49	0.5830	21.70	0.5710	23.16	0.7250
LapSRN [12]	×8	26.15	0.7380	24.35	0.6200	24.54	0.5860	21.81	0.5810	23.39	0.7350
MemNet [24]	×8	26.16	0.7414	24.38	0.6199	24.58	0.5842	21.89	0.5825	23.56	0.7387
EDSR [16]	×8	26.96	0.7762	24.91	0.6420	24.81	0.5985	22.51	0.6221	-/-	-/-
MSRN [15]	×8	26.59	0.7254	24.88	0.5961	24.70	0.5410	22.37	0.5977	24.28	0.7517
HRAN (ours)	×8	27.11	0.7798	25.01	0.6419	24.83	0.5983	22.57	0.6223	24.64	0.7817
HRAN+ (ours)	×8	27.18	0.7828	25.12	0.6450	24.89	0.6001	22.73	0.6280	24.87	0.7878

TABLE 2. Investigation of HRAB module (with and without CA). We examine the best PSNR (dB) on Urban100 (2×) with same training settings.

Modules	SSIM / PSNR
SA without CA	32.77 / 0.9343
SA with CA (Hybrid Attention)	32.95 / 0.9357

TABLE 3. BFF vs HFF structures. We examine the best PSNR (dB) on Set5 (2×) with same training settings.

Method	PSNR / SSIM
MSRN with HFF [15]	32.22 / 0.9326
MSRN [15] with proposed BFF	32.44 / 0.9315
Our HRAN with HFF	32.69 / 0.9334
Our HRAN with proposed BFF	32.95 / 0.9375

dataset [9] as it consists of a large dataset. The results are shown in Table 2. We observe that our SA module alone achieves 32.77 dB PSNR. We also experiment with CA module only though results were unsatisfactory. Whereas, when we combine SA with CA, i.e. our HRAB module, it achieves the 32.95 dB PSNR. This study suggests we need HRAB module containing both the spatial and channel attention for accurate SR results. We also investigate about our BFF structure using HRAB module and tested the both BFF and HFF on MSRN [15] and proposed HRAN to verify the effectiveness of BFF on both models. It is evident from the results that BFF structure improves the PSNR of MSRN [15] from 32.22 dB to 32.44 dB by just replacing the HFF with BFF. Moreover, proposed HRAN and BFF together significantly increase the accuracy which show the effectiveness of our BFF structure.

C. MODEL COMPLEXITY ANALYSIS

Since we are targeting the maximum accuracy with limited memory computation, therefore our performance is best visible when we see the Table.1 along with Figure. 4. In Figure. 4, we compare our model size and its performance on Urban100 dataset with scale factor 4 as it is more difficult dataset as compared to others, and also it consists of large number of images. As we observe that our HRAN model has fewer parameters compared to RDN [36] and EDSR [16], nevertheless it still achieves the comparable performance whereas our HRAN+ outperforms the state-of-the-art methods. For the fair comparisons, we trained the high-capacity model of MSRN [15] which contains 50% more residual blocks, denoted with MSRN* in the Figure 4. We can see that our model still achieves better performance than MSRN* with lower number of parameters. Thus, these results demonstrate the effective utilization of the features that result in performance gain in SISR.

V. CONCLUSION

In this paper, we propose the HRAN to detect the most informative multiscale spatial features for the accurate SR image. The proposed HRAB module fully utilizes the high-frequency information from input features with a combination of the SA and CA. In addition, the proposed BFF structure allows us to smoothly transmit all the features at the end of the network for reconstruction. Furthermore, we propose the adoption of the GSC, SSC, LSC and GR to ease the flow of information through the network. Our comprehensive experiments show the efficacy of the proposed model.

REFERENCES

[1] M. Bevilacqua, A. Roumy, C. Guillemot, and M. Alberi-Morel, "Low-complexity single-image super-resolution based on nonnegative neighbor embedding," in *Proc. Brit. Mach. Vis. Conf. (BMVC)*, 2012, pp. 1–10.
[2] C.-F. Chen, Q. Fan, N. Mallinar, T. Sercu, and R. Feris, "Big-Little net: An efficient multi-scale feature representation for visual and speech recognition," 2018, *arXiv:1807.03848*. [Online]. Available: https://arxiv.org/abs/1807.03848
[3] C. Dong, C. C. Loy, K. He, and X. Tang, "Image super-resolution using deep convolutional networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, no. 2, pp. 295–307, Feb. 2015.
[4] C. Dong, C. C. Loy, and X. Tang, "Accelerating the super-resolution convolutional neural network," in *Proc. Eur. Conf. Comput. Vis.* Springer, 2016, pp. 391–407.
[5] V. Dumoulin, J. Shlens, and M. Kudlur, "A learned representation for artistic style," in *Proc. ICLR*, vol. 2, 2017.
[6] M. Haris, G. Shakhnarovich, and N. Ukita, "Deep back-projection networks for super-resolution," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2018, pp. 1664–1673.
[7] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.
[8] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2018, pp. 7132–7141.
[9] J.-B. Huang, A. Singh, and N. Ahuja, "Single image super-resolution from transformed self-exemplars," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 5197–5206.
[10] J. Kim, J. K. Lee, and K. M. Lee, "Accurate image super-resolution using very deep convolutional networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016.
[11] J. Kim, J. K. Lee, and K. M. Lee, "Deeply-recursive convolutional network for image super-resolution," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 1637–1645.
[12] W.-S. Lai, J.-B. Huang, N. Ahuja, and M.-H. Yang, "Deep Laplacian pyramid networks for fast and accurate super-resolution," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 624–632.
[13] C. Lanaras, E. Baltsavias, and K. Schindler, "Hyperspectral super-resolution by coupled spectral unmixing," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 3586–3594.
[14] C. Ledig, L. Theis, F. Huszar, J. Caballero, A. Cunningham, A. Acosta, A. Aitken, A. Tejani, J. Totz, Z. Wang, and W. Shi, "Photo-realistic single image super-resolution using a generative adversarial network," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 4681–4690.
[15] J. Li, F. Fang, K. Mei, and G. Zhang, "Multi-scale residual network for image super-resolution," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, Sep. 2018, pp. 517–532.
[16] B. Lim, S. Son, H. Kim, S. Nah, and K. M. Lee, "Enhanced deep residual networks for single image super-resolution," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 136–144.
[17] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 2117–2125.
[18] D. Martin, C. Fowlkes, D. Tal, and J. Malik, "A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics," in *Proc. 8th IEEE Int. Conf. Comput. Vis. (ICCV)*, Jul. 2001, pp. 416–423.

[19] Y. Matsui, K. Ito, Y. Aramaki, A. Fujimoto, T. Ogawa, T. Yamasaki, and K. Aizawa, "Sketch-based manga retrieval using manga109 dataset," *Multimedia Tools Appl.*, vol. 76, no. 20, pp. 21811–21838, 2017.
[20] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "MobileNetV2: Inverted residuals and linear bottlenecks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2018, pp. 4510–4520.
[21] S. Seferbekov, V. Iglovikov, A. Buslaev, and A. Shvets, "Feature pyramid network for multi-class land segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2018, pp. 272–2723.
[22] W. Shi, J. Caballero, F. Huszár, J. Totz, A. P. Aitken, R. Bishop, D. Rueckert, and Z. Wang, "Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 1874–1883.
[23] W. Shi, J. Caballero, C. Ledig, X. Zhuang, W. Bai, K. Bhatia, A. M. S. M. de Marvao, T. Dawes, D. O'Regan, and D. Rueckert, "Cardiac image super-resolution with global correspondence using multi-atlas patchmatch," in *Proc. Int. Conf. Med. Image Comput. Comput.-Assist. Intervent.*, Springer, 2013, pp. 9–16.
[24] Y. Tai, J. Yang, X. Liu, and C. Xu, "MemNet: A persistent memory network for image restoration," in *Proc. Int. Conf. Comput. Vis.*, Oct. 2017, pp. 4539–4547.
[25] W. Tan, B. Yan, and B. Bare, "Feature super-resolution: Make machine see more clearly," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2018, pp. 3994–4002.
[26] R. Timofte, E. Agustsson, L. Van Gool, M.-H. Yang, and L. Zhang, "NTIRE 2017 challenge on single image super-resolution: Methods and results," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 114–125.
[27] R. Timofte, R. Rothe, and L. Van Gool, "Seven ways to improve example-based single image super resolution," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR),*, Jun. 2016, pp. 1865–1873.
[28] T. Tong, G. Li, X. Liu, and Q. Gao, "Image super-resolution using dense skip connections," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 4799–4807.
[29] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: From error visibility to structural similarity," *IEEE Trans. Image Process.*, vol. 13, no. 4, pp. 600–612, Apr. 2004.
[30] Z. Wang, D. Liu, J. Yang, W. Han, and T. Huang, "Deep networks for image super-resolution with sparse prior," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 370–378.
[31] J. Yang, J. Wright, T. S. Huang, and Y. Ma, "Image super-resolution via sparse representation," *IEEE Trans. Image Process.*, vol. 19, no. 11, pp. 2861–2873, Nov. 2010.
[32] F. Yu, V. Koltun, and T. Funkhouser, "Dilated residual networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jul. 2017, pp. 472–480.
[33] R. Zeyde, M. Elad, and M. Protter, "On single image scale-up using sparse-representations," in *Proc. Int. Conf. Curves Surf.* Springer, 2010, pp. 711–730.
[34] K. Zhang, W. Zuo, and L. Zhang, "Learning a single convolutional super-resolution network for multiple degradations," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 3262–3271.
[35] Y. Zhang, K. Li, K. Li, L. Wang, B. Zhong, and Y. Fu, "Image super-resolution using very deep residual channel attention networks," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 286–301.
[36] Y. Zhang, Y. Tian, Y. Kong, B. Zhong, and Y. Fu, "Residual dense network for image super-resolution," in *Proc. CVPR*, Jun. 2018, pp. 2472–2481.
[37] Z. Zhang, X. Wang, and C. Jung, "DCSR: Dilated convolutions for single image super-resolution," *IEEE Trans. Image Process.*, vol. 28, no. 4, pp. 1625–1635, Apr. 2019.
[38] W. W. W. Zou and P. C. Yuen, "Very low resolution face recognition problem," *IEEE Trans. Image Process.*, vol. 21, no. 1, pp. 327–340, Jan. 2012.
[39] C. Wang, Z. Li, and J. Shi, "Lightweight image super-resolution with adaptive weighted learning network," 2019, *arXiv:1904.02358*. [Online]. Available: https://arxiv.org/abs/1904.02358
[40] X. Sun, W. Lu, R. Wang, and F. Bai, "Distilling with residual network for single image super resolution," In *Proc. IEEE Int. Conf. Multimedia Expo (ICME)*, Jul. 2019, pp. 1180–1185.
[41] P. Liu, H. Zhang, W. Lian, and W. Zuo, "Multi-level wavelet convolutional neural networks," *IEEE Access*, vol. 7, pp. 74973–74985, 2019.

ABDUL MUQEET received the bachelor's degree from the University of Karachi, Karachi, Pakistan, in 2014. He is currently pursuing the M.S. degree leading to the Ph.D. Program with Kyung Hee University, South Korea. His research interests include deep learning interpretation and efficient deep learning models.

MD TAUHID BIN IQBAL received the bachelor's degree in information technology from the University of Dhaka, in December 2012, and the Ph.D. degree from the Department of Computer Science and Engineering, Kyung Hee University, South Korea, in February 2019, where he is currently a Postdoctoral Fellow with the Machine Learning and Visual Computing Lab. His research interests include deep learning interpretation, efficient deep learning architecture design, facial analysis, including expression recognition, combined age, and gender recognition.

SUNG-HO BAE received the B.S. degree from Kyung Hee University, South Korea, in 2011, and the M.S. and Ph.D. degrees from the Korea Advanced Institute of Science and Technology (KAIST), Daejeon, South Korea, in 2012 and 2016, respectively. From 2016 to 2017, he was a Postdoctoral Associate with the Computer Science and Artificial Intelligence Laboratory (CSAIL), Massachusetts Institute of Technology (MIT), MA, USA. Since 2017, he has been an Assistant Professor with the Department of Computer Science and Engineering, Kyung Hee University. He has been involved in model compression/interpretation for deep neural networks and inverse problems in image processing and computer vision.