A Two-Branch Multiscale Residual Attention Network for Single Image Super-Resolution in Remote Sensing Imagery

Allen Patnaik, M. K. Bhuyan, *Senior Member, IEEE*, and Karl F. MacDorman

***Abstract*—High-resolution remote sensing imagery finds applications in diverse fields, such as land-use mapping, crop planning, and disaster surveillance. To offer detailed and precise insights, reconstructing edges, textures, and other features is crucial. Despite recent advances in detail enhancement through deep learning, disparities between original and reconstructed images persist. To address this challenge, we propose a two-branch multiscale residual attention network for single-image super-resolution reconstruction. The network gathers complex information about input images from two branches with convolution layers of different kernel sizes. The two branches extract both low-level and high-level features from the input image. The network incorporates multiscale efficient channel attention and spatial attention blocks to capture channel and spatial dependencies in the feature maps. This results in more discriminative features and more accurate predictions. Moreover, residual modules with skip connections can help to overcome the vanishing gradient problem. We trained the proposed model on the WHU-RS19 dataset, collated from Google Earth satellite imagery, and validated it on the UC Merced, RSSCN7, AID, and real-world satellite datasets. The experimental results show that our network uses features at different levels of detail more effectively than state-of-the-art models.**

***Index Terms*—Attention, deep learning, high-resolution (HR), low-resolution (LR), residual connection.**

I. Introduction

Satellite images have various applications, including agricultural monitoring [1], [2], climate monitoring [3], environmental analysis [4], and surveillance [5]. However, their spatial resolution is limited. To address this, researchers have proposed various algorithms for single-image super-resolution (SISR) reconstruction to generate a high-quality, high-resolution (HR) image from a low-resolution (LR) source image.

SISR methods can be broadly divided into classical and deep learning. Classical methods include interpolation-based, such as bicubic and bilinear; edge-based, such as the Canny edge detector and reconstruction-based such as total variation regularization. Classical SISR methods face challenges in generating high-quality, detailed images. Deep learning methods train deep neural networks, such as convolution neural networks (CNNs) and generative adversarial networks (GANs) on LR and HR image pairs to learn a mapping from LR to HR images. These methods are widely employed to produce high-quality, detailed images.

Deep learning architectures have advanced over the past decade, improving SISR accuracy. Dong et al. [6] introduced a simple three-layer convolutional network to process natural images. This structure produced higher quality images than earlier methods. However, it still lacked deep feature extraction, essential for accurate image reconstruction. To extract deep features, Tong et al. [7] proposed DenseNet, a CNN featuring skip connections and residual blocks with subpixel convolutions for upscaling. Although DenseNet extracts deep features, it consumes more memory and other resources. Lim et al. [8] reduced memory consumption by removing the batch normalization layer without reducing reconstruction performance. However, the network lacked discriminative learning because it weighs channel features equally.

Residual blocks that aggregate feature components have been employed to enhance SR performance [9], [10], [11] yielding promising results though still failing to restore high-frequency image details. To address this, attention blocks have been introduced to capture additional information to improve reconstruction [12], [13], [14]. However, these networks struggled with the complex textures found in remote-sensing images. In response, researchers have developed targeted approaches to improve SISR reconstruction. These include extracting higher-order statistics [15], employing wavelet decomposition [16], and using transformers [17]. Yet, to maintain processing speed, most models still ignore the multiscale features of LR images despite their importance for accurate reconstruction.

To detect image features across scales, some researchers have adopted multiscale structures with convolution kernels of varying sizes. Cao and Liu [18] introduced a network that integrates channel attention (CA) with multiscale features to enhance image details. Lei and Liu [19] proposed a network that captures multiscale features by integrating the inception, spatial attention (SA), and residual models. Huan et al. [20] devised a pyramidal network with blocks for multiscale dilation convolution. Zhang et al. [21] designed a dual-resolution model, which merges spatial information using two separate branches. Wang et al. [22] applied a scheme for omni-dimension feature

Manuscript received 14 April 2023; revised 8 August 2023, 8 December 2023, and 20 January 2024; accepted 25 February 2024. Date of publication 29 February 2024; date of current version 13 March 2024. *(Corresponding author: Allen Patnaik.)*

Allen Patnaik and M. K. Bhuyan are with the Department of Electronics and Electrical Engineering, Indian Institute of Technology Guwahati, Guwahati 781039, India (e-mail: allen.patnaik@iitg.ac.in; mkb@iitg.ac.in).

Karl F. MacDorman is with the Luddy School of Informatics, Computing and Engineering, Indiana University, Bloomington, IN 46202 USA (e-mail: kmacdorm@indiana.edu).

Digital Object Identifier 10.1109/JSTARS.2024.3371710

aggregation to capture scale patterns in different dimensions. Wang et al. [23] incorporated a residual block with fast Fourier transforms to recover information at various scales. Although these networks reconstruct details at multiple scales, fully integrating features from both spatial and channel domains is also required.

Researchers have begun to innovate with models reflecting real-world imagery. Dong et al. [24] developed a model introducing image degradation by simulating noise and blur. However, it only used RGB bands and applied JPEG and simple noise compression. Qiu et al. [25] improved their network architecture by incorporating blind noise and blur kernel models, enabling the extraction of edge features with edge filters. However, their method does not yet accommodate various target sensors. Xiao et al. [26] created a dual feature modulation network that addresses multiple degradation models, employing a contrastive learning strategy to refine real-world imagery.

Despite these efforts to create a deep learning model for enhancing remote sensing images, many current approaches still fall short. They do not consider the different features of objects and thus fail to represent their underlying structure. Texture blur and spectral distortion become severe when using a higher upscaling factor. To solve these problems, we designed a two-branch multi-scale residual attention (TBMRA) network for SISR reconstruction.

This article makes the following contributions.

1) A new TBMRA module consisting of two parallel branches of convolution networks with different kernel sizes is employed to extract information from input images.
2) The module incorporates a hybrid multiscale attention mechanism to extract global features from the image's region of interest. Our proposed attention module uses different kernel sizes for extracting image features at different scales.
3) We evaluated the impact of different attention mechanisms and kernel sizes using peak signal-to-noise ratio (PSNR) and structure similarity index measurement (SSIM). Extensive experiments comparing our method to state-of-the-art approaches on synthetic and nonsynthetic datasets favored our method.

The rest of this article is organized as follows. Section II reviews related work. Section III describes the proposed method and network architecture. Section IV presents experimental results comparing the performance of our network architecture to the state-of-the-art. Finally, Section V concludes this article.

II. Related Work

A. Feature Extraction Block

Different types of feature extraction blocks have been proposed, including inception, residual, residual dense, and multiscale residual. Inception blocks [27] use different-sized convolution layers with a max pooling layer to extract features. Residual blocks [28] use skip connections where inputs are added to the output without passing through the whole network. Residual dense blocks [9] use convolution layers with dense connections to extract local and global features. Multiscale residual blocks [29] are concatenated with convolution blocks with skip connections to explore features more thoroughly. This article uses multiscale residual blocks to extract the features.

B. CNN-Based SISR for Remote Sensing Images

CNNs have gained prominence in super-resolution image reconstruction for remote sensing, which often suffers from limited spatial resolution due to sensor limitations, atmospheric conditions, or satellite altitude. As a result, models designed to enhance the resolution of natural images may fall short in remote sensing applications.

Many researchers have successfully employed attention-based methodologies [15], [23], [30]. Advances include the use of residual dense networks and several novel techniques that reduce model complexity [16], [31], [32]. Researchers have utilized self-supervised [26] or self-similarity [33] approaches to group adjacent image regions or similar objects. Also, GAN methods have produced high-quality super-resolved images [34], [35]. Despite progress, the quest for more robust models persists. Our model draws inspiration from prior research by combining residual and attention-based techniques.

III. Method and Architecture

This section describes our network's structure and attention mechanism, shown in Fig. 2(a)–(c). Its backbone is a multiscale residual network. The attention mechanism combines multiscale efficient channel and SA to obtain more discriminative features.

A. Network Architecture

As shown in Fig. 1, we extract features of LR images using the initial convolution operation with a kernel size of 3 × 3 defined by

$$F_0 = w_{3\times 3} * (I_{\mathrm{LR}}) \tag{1}$$

where F_0 denotes the output features obtained by the convolution layer, and I_{LR} denotes the LR image.

To extract intermediate features, the extracted features pass through the TBMRA block. This process can be formulated as

$$F_1 = (H_{D0}, H_{D1}, \ldots, H_{Dn})(F_0) \tag{2}$$

where F_1 denotes the output features of the TBMRA block, and H_D denotes the multi $\backslash$ scale residual attention block.

Global residual learning: We used *N* blocks of global residual learning in the TBMRA block to preserve important features of the whole image. This process can be described as

$$F_g = F_1 + F_{g-1} \tag{3}$$

where F_g and F_{g-1} denote the global and initial feature maps.

Next, the extracted deep features pass through a bottleneck layer with a 1 × 1 kernel size convolution layer to compress the feature representation of the structure. The result is

$$F_2 = H_{\mathrm{B}}(F_g) \tag{4}$$

where F_2 denotes the output of the bottleneck structure and H_{B} denotes the bottleneck layer.

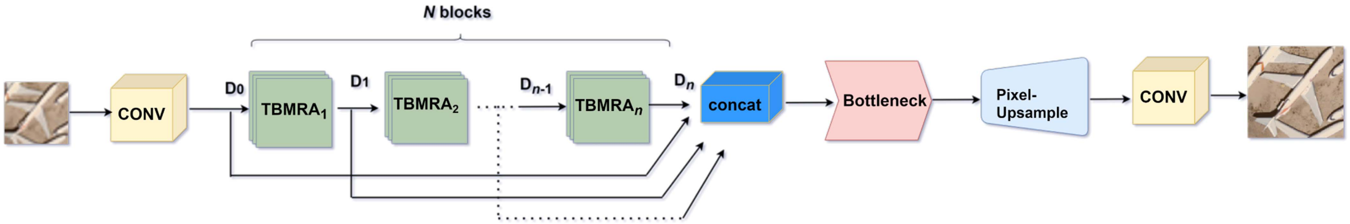


Fig. 1. Proposed network architecture. *N* blocks of residual networks with 3 × 3 convolution kernel sizes are used.

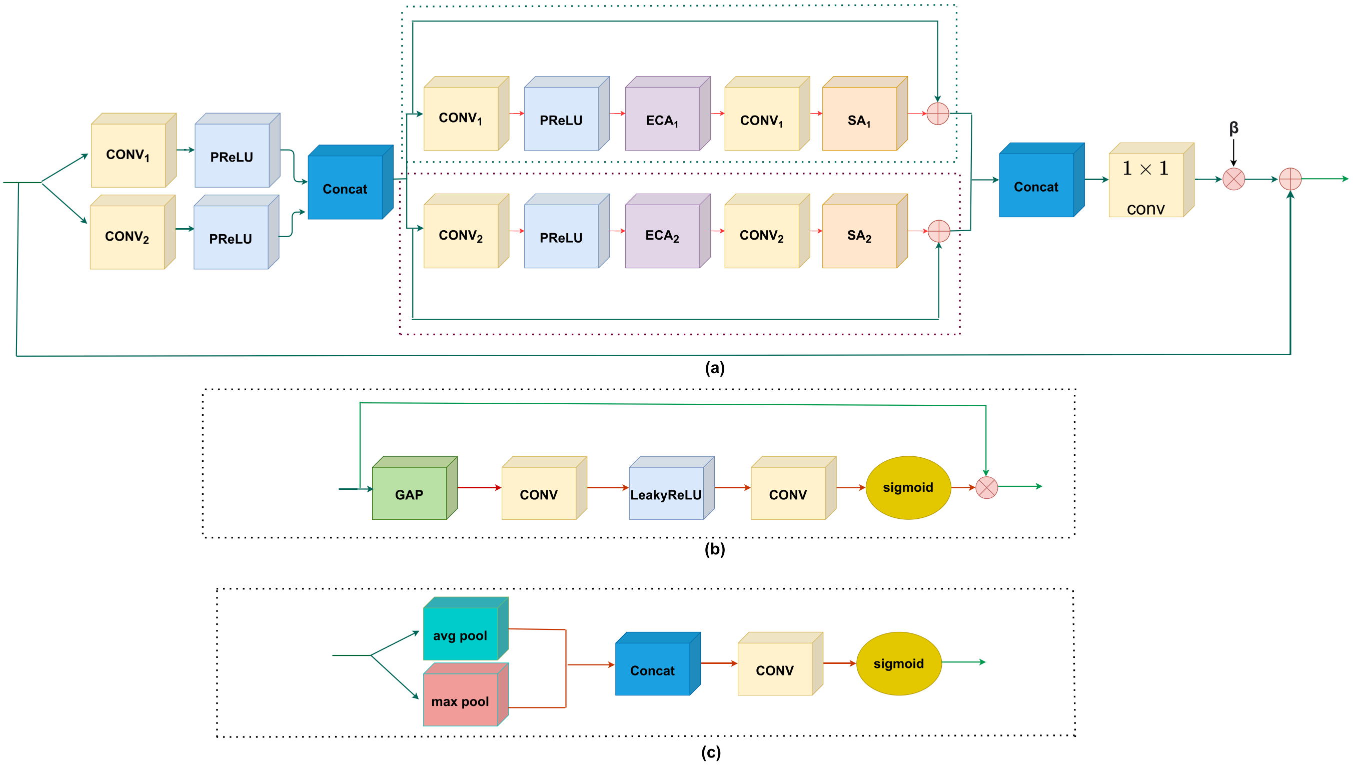


Fig. 2. (a) Overview of the proposed TBMRA module. CONV$_1$ and CONV$_2$ have 3 × 3 and 5 × 5 convolution kernel sizes, respectively. (b) Architecture of ECA with two 3 × 3 convolution sizes for the upper branch and two 5 × 5 convolution sizes for the lower branch. (c) architecture of SA with 3 × 3 for upper branch and 5 × 5 for lower branch.

Pixel shuffle [36] is used to upsample these deep features, which increase the image's resolution by increasing its height and width

$$F_3 = H_{\text{UP}}(F_2) \tag{5}$$

where F_3 denotes the increased spatial dimension of the layer and H_{UP} denotes the pixel shuffle upsampling layer.

Finally, we obtain the super-resolved HR image by passing this result through the last convolution layer

$$I_{\text{SR}} = w_{3\times 3} * (F_3) \tag{6}$$

where I_{SR} denotes the final reconstruction features of the super-resolved image after the last convolution layer.

The network is designed to make the super-resolved image (I_{SR}) closer to the HR image (I_{HR}). It is trained using an L1 Charbonnier loss function to achieve this. This can be described as

$$L(\theta_i) = \frac{1}{N_i}\sum_{i=1}^{N_i} \rho||H_{\text{o}}(I_{\text{SR}}^i) - I_{\text{HR}}^i|| \tag{7}$$

where θ_i denotes the parameters in the whole network, N_i denotes the number of training samples, H_{o} denotes the TBMRA model, and ρ denotes the Charbonnier penalty function.

B. Hybrid Multiscale Attention

Attention has emerged as a recent advancement in convolution architectures. Different attention mechanisms have proved useful for generating feature maps. The attention process begins with the squeeze and excitation (SE) operation, which takes the SE module as its building block for recalibrating feature maps. The algorithm is described below.

Let a convolution block's input feature map be $\chi_c \in \mathbb{R}^{W \times H \times C}$, describing the width, height, and channel information. The channel's weight in the block can be computed as

$$\omega_c = \phi(f_{\{W_{\text{se1}}, W_{\text{se2}}\}}(g(\chi_c))) \tag{8}$$

where $g(\chi_c) = \frac{1}{W_c H_c} \sum_{i=1,j=1}^{W_c H_c} \chi_{i,j}$ is the global average pooling (GAP), and ϕ is a sigmoid function. $f_{\{W_{\text{se1}}, W_{\text{se2}}\}}$ includes all the parameters of the SE block.

The weights of $W1$ and $W2$ are calculated as

$$W1 = \begin{bmatrix} w_{1,1} & \cdots & w_{1,\frac{c}{r}} \\ \vdots & \ddots & \vdots \\ w_{c,1} & \cdots & w_{c,\frac{c}{r}} \end{bmatrix} \quad W2 = \begin{bmatrix} w_{1,1} & \cdots & w_{1,c} \\ \vdots & \ddots & \vdots \\ w_{\frac{c}{r},1} & \cdots & w_{\frac{c}{r},c} \end{bmatrix} \tag{9}$$

where r is the reduction parameter.

Efficient channel attention (ECA) [37] follows the SE module to capture information between the channels. It increases the architecture's efficiency and effectiveness by reducing channel dependencies. The ECA module's weights are represented by the weight matrix $W3$

$$W3 = \begin{bmatrix} w_{1,1} & \cdots & w_{1,\text{ks}} & 0 & 0 & \cdots & \cdots & 0 \\ 0 & w_{2,2} & \cdots & w_{2,\text{ks}} & \cdots & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & \cdots & 0 & 0 & \cdots & w_{c,c-\text{ks}+1} & \cdots & w_{c,c} \end{bmatrix} \tag{10}$$

where ks denotes the kernel size of the weight matrix. The $W3$ weight matrix is sparser than $W1$ and $W2$.

The ECA approach can be represented with a simple 1-D convolution kernel with a kernel size of ks

$$\omega_{\text{eca}} = \phi(\text{conv}_{ks}^{\text{1-D}}(\text{Input})). \tag{11}$$

We propose to use hybrid multiscale attention to apply different kernels to the channel and SA maps.

Let *I* be the input feature maps and M_c and M_s be the channel and SA maps, respectively. Then, the output of the module can be represented as

$$O = I * M_c * M_s \tag{12}$$

where $*$ denotes elementwise multiplication. The CA map M_c and the SA map M_s are computed as

$$M_c = \phi(f(\text{AvgPool}(X))) \tag{13}$$

$$M_s = \phi(f(\text{Conv1D}(X))) \tag{14}$$

where AvgPool denotes the GAP operation, Conv1D denotes a convolutional block with a 1×1 kernel size, f denotes a multilayer perceptron, and ϕ denotes the sigmoid activation function.

An ECA map is used to exploit the efficient and effective interchannel relation among features, as shown in Fig. 2(b). The map extracts useful information from channels by applying GAP. The squeezed 1-D feature map $F_c \in c \times 1 \times 1$ then passes through two convolution layers with kernel sizes of 3 and 5. In between them, a leaky ReLU activation function is used to represent the nonlinearity with a small, nonzero slope for the negative part of the function. The sigmoid function can compute the ECA map

$$M_{\text{ec}}(F) = \phi(W5\text{LeakyReLU}(W4(F_{\text{avg}}^c))). \tag{15}$$

A spatial map can be produced to exploit the intraspatial relations among the features, as shown in Fig. 2(c). Two types of pooling are used, GAP and global max pooling, which produce two 2-D feature maps. These 2-D feature maps are concatenated and passed through 2-D convolutions of kernel sizes 3 and 5. Next, the sigmoid convolution generated a SA map

$$M_s(F) = \phi((W6)(F_{\text{avg}}^{\text{s}}; F_{\text{max}}^{\text{s}})). \tag{16}$$

C. TBMRA Module

The TBMRA module consists of a multiscale residual block, ECA, and SA attention. We use a residual structure to obtain a deeper network with ease of training by bypassing the information in intermediate layers with a skip connection.

The multiscale residual block fuses information extracted from the images at various levels by adapting convolutional kernels of different sizes, as shown in Fig. 2(a). The top–bottom layers employ two convolutional blocks with kernel sizes of 3×3 and 5×5. The network's first branch processes the LR input image, extracting low-level features, such as edges and contours, to capture local features. The second branch processes the same image, extracting high-level features, such as textures and patterns, to capture global features. The parametric ReLU (PReLU) activation function with a learnable slope for the negative part of the function is applied to both layers, introducing nonlinearity into the model.

Let the initial features be denoted as $F \in \mathbb{R}^{W \times H \times C}$.

G_0 and G_1 are the output of initial feature extraction with 3×3 and 5×5 kernel sizes for convolution

$$G_0 = \sigma_p(w_{3\times3} * (F)) \tag{17}$$

$$G_1 = \sigma_p(w_{5\times5} * (F)) \tag{18}$$

where $\sigma_p(\cdot)$ denotes the PReLU activation function.

C is the concatenated features along the channel dimension.

$$C = [G_0, G_1]. \tag{19}$$

Next, the network divides into two branches, as shown in Fig. 2(a), with a convolution layer of 3×3 kernel size for the upper branch and 5×5 for the lower branch, obtaining G_2 and G_3 for the outputs, respectively

$$G_2 = \sigma_p(w_{3\times3} * (C)) \tag{20}$$

$$G_3 = \sigma_p(w_{5\times5} * (C)). \tag{21}$$

Then, the upper branch flows through ECA with two kernels of sizes 3×3, and the lower branch through ECA with two kernels of sizes 5×5, obtaining G_4 and G_5 as the outputs

$$G_4 = M_{ec3\times3}(G_2) \tag{22}$$

$$G_5 = M_{ec5\times5}(G_3) \tag{23}$$

TABLE I
PSNR, SSIM, AND LPIPS VALUES OF DIFFERENT METHODS FOR SCALE ×2 ON THREE DATASETS

Dataset	Metric	Bicubic	OmniSR	DCM	MHAN	EDSR	RFDN	CTNET	ESRGAN	TBMRA
UC Merced	PSNR	30.600	34.776	34.847	34.739	34.833	34.758	34.236	34.873	**35.001**
	SSIM	0.8544	0.922	0.929	0.920	0.927	0.925	0.918	0.932	**0.936**
	LPIPS	0.057	0.038	0.033	0.040	0.034	0.035	0.043	0.032	**0.030**
RSSCN7	PSNR	29.760	32.523	32.542	32.478	32.539	32.480	32.367	32.629	**32.835**
	SSIM	0.868	0.890	0.894	0.888	0.892	0.891	0.885	0.896	**0.901**
	LPIPS	0.069	0.055	0.051	0.052	0.052	0.053	0.056	0.050	**0.048**
AID	PSNR	31.342	36.778	36.945	36.684	36.787	36.744	36.496	36.851	**37.176**
	SSIM	0.862	0.942	0.944	0.940	0.943	0.943	0.937	0.946	**0.950**
	LPIPS	0.052	0.028	0.026	0.028	0.026	0.029	0.032	0.025	**0.024**

The bold values denote the best results.

TABLE II
PSNR, SSIM, AND LPIPS VALUES OF DIFFERENT METHODS FOR SCALE ×3 ON THREE DATASETS

Dataset	Metric	Bicubic	OmniSR	DCM	MHAN	EDSR	RFDN	CTNET	ESRGAN	TBMRA
UC Merced	PSNR	26.320	31.629	31.638	31.616	31.636	31.594	30.139	30.496	**31.773**
	SSIM	0.8051	0.847	0.847	0.846	0.847	0.846	0.840	0.855	**0.859**
	LPIPS	0.081	0.068	0.066	0.065	0.069	0.069	0.073	0.069	**0.065**
RSCNN7	PSNR	25.860	29.787	29.789	29.733	29.781	29.751	29.607	29.778	**29.907**
	SSIM	0.801	0.810	0.809	0.809	0.809	0.808	0.803	0.809	**0.813**
	LPIPS	0.133	0.110	0.108	0.111	0.110	0.115	0.124	0.111	**0.101**
AID	PSNR	27.401	32.862	32.825	32.765	32.804	32.679	32.511	32.826	**33.048**
	SSIM	0.825	0.878	0.878	0.876	0.877	0.874	0.871	0.877	**0.882**
	LPIPS	0.088	0.067	0.064	0.065	0.065	0.068	0.080	0.067	**0.060**

The bold values denote the best results.

where $M_{ec3\times3}$ and $M_{ec5\times5}$ denote the ECA map for 3 × 3 and 5 × 5 kernel sizes.

Next, the lower and upper branches repeat the convolution layers of 3 × 3 and 5 × 5 kernel sizes, obtaining G_6 and G_7 as the outputs

$$G_6 = w_{3\times3}(G_4) \tag{24}$$

$$G_7 = w_{5\times5}(G_5). \tag{25}$$

Finally, SA of kernel sizes 3 × 3 and 5 × 5 are used for the lower and upper branches, obtaining G_8 and G_9 as the outputs, respectively

$$G_8 = M_{s3\times3}(G_6) \tag{26}$$

$$G_9 = M_{s5\times5}(G_7) \tag{27}$$

where $M_{s3\times3}$ and $M_{s5\times5}$ are the SA maps for 3 × 3 and 5 × 5 kernel sizes.

Local residual learning: To enhance information retention, we employ local residual learning within the feature extraction block, merging the initial and final output. Formally, we describe it as

$$I_n = G_{\mathrm{U/B}} + I_{n-1} \tag{28}$$

where I_{n-1} and I_n represent the block's input and output, respectively. $G_{\mathrm{U/B}}$ is the output feature of the upper or lower branch of the network.

The output is then concatenated and passed through the convolution layer of kernel size 1×1 to reduce the channel depth, obtaining H_{D} as the output

$$H_{\mathrm{D}} = w^{1\times1}[G_8, G_9] \tag{29}$$

The output feature map is multiplied by a residual scaling parameter β to increase training stability [8].

Finally, the original inputs are added to the output through residual connections to obtain the HR image.

TABLE III
PSNR, SSIM, and LPIPS Values of Different Methods for Scale ×4 on Three Datasets

Dataset	Metric	Bicubic	OmniSR	DCM	MHAN	EDSR	RFDN	CTNET	ESRGAN	TBMRA
UC Merced	PSNR	25.050	29.727	29.836	29.884	29.912	29.925	28.812	29.966	**30.141**
	SSIM	0.688	0.765	0.769	0.771	0.772	0.772	0.763	0.775	**0.787**
	LPIPS	0.137	0.115	0.103	0.096	0.100	0.102	0.103	0.102	**0.092**
RSCNN7	PSNR	24.320	28.162	28.274	28.314	28.321	28.235	28.117	28.340	**28.512**
	SSIM	0.715	0.726	0.732	0.733	0.734	0.731	0.724	0.735	**0.743**
	LPIPS	0.184	0.166	0.155	0.149	0.150	0.157	0.159	0.152	**0.137**
AID	PSNR	25.745	30.262	30.401	30.452	30.498	30.373	30.141	30.540	**30.837**
	SSIM	0.782	0.796	0.802	0.804	0.806	0.801	0.794	0.806	**0.816**
	LPIPS	0.143	0.122	0.112	0.102	0.106	0.119	0.114	0.105	**0.089**

The bold values denote the best results.

TABLE IV
Quantitative Comparison of the Models by Varying Different Types of Attention Modules (Scaling Factor ×4)

Attention Modules	PSNR	SSIM
Without Attention	28.416	0.737
CA	28.451	0.739
SA	28.449	0.738
ECA	28.465	0.739
CA+SA	28.493	0.741
ECA+SA	**28.512**	**0.743**

The bold values denote the best results.

TABLE V
Comparison of Our Model With Two Kernel Sizes (Scaling Factor ×4)

Kernel sizes	PSNR	SSIM
×3 ; ×5	**28.512**	**0.743**
×5 ; ×7	28.502	0.741
×3 ; ×7	28.506	0.741

The bold values denote the best results.

TABLE VI
Quantitative Comparison of the Models by Varying the Numbers of Residual Blocks "*N*" (Scaling Factor × 4)

Residual number	5	10	15	20
	PSNR / SSIM	PSNR / SSIM	PSNR / SSIM	PSNR / SSIM
EDSR	28.321 / 0.734	28.345 / 0.735	28.352 / 0.736	28.359 / 0.736
RFDN	28.235 / 0.731	28.237 / 0.732	28.241 / 0.733	28.248 / 0.733
ESRGAN	28.340 / 0.735	28.343 / 0.735	28.356 / 0.737	28.360 / 0.737
TBMRA	28.512 / 0.743	28.513 / 0.743	28.533 / 0.744	**28.537 / 0.744**

The bold values denote the best results.

IV. Experiments

This section describes the datasets, training parameters, evaluation metrics, and loss functions used to verify our model's effectiveness. We also compare our model with other methods.

A. Datasets

We used the WHU-RS19 dataset for training and UC Merced, RSSCN7, and AID datasets for testing. WHU-RS19 [38], [39] is a set of 1005 Google Earth satellite images compiled by Wuhan University. Each image is 600 × 600 with a pixel resolution of up to 0.5 m. The set is divided into 19 classes of scenes with about 50 samples per class. The dataset includes airports, beaches, football fields, parking lots, and residential areas. To train our model, we randomly selected 80%, i.e., 800 images, from the WHU-RS19 dataset for each class. We used the remaining 20% of images from each class to validate the model.

The UC Merced dataset [40] is a collection of 2100 images, each measuring 256 × 256 pixels. Images with a spatial resolution of one foot were manually selected from the USGS National Map Urban Area Imagery collection. The dataset includes agriculture, airplanes, baseball diamonds, beaches, and buildings. RSSCN7 [41] is a public dataset released by Wuhan University in 2015. The dataset contains 400 × 400 pixel images of 2800 image samples. The images include grassland, farmland, industrial and commercial regions, rivers and lakes, forest fields, residential regions, and parking lots. AID [42] is a collection of scene classification remote sensing images from Google Earth. This dataset consists of 600 × 600 pixel images of 2400 image samples. The test set consisted of randomly chosen images from each class of the UC Merced, RSSCN7, and AID datasets.

Furthermore, we validate our model with real-world satellite data consisting of 4123 GaoFen-2 (GF-2) [43] images. The spatial resolution is 0.8 m with 480 × 480 pixels of each sample image.

B. Training Parameters

Using bicubic interpolation, the model images were downsampled from their originals by a factor of ×2, ×3, and ×4.

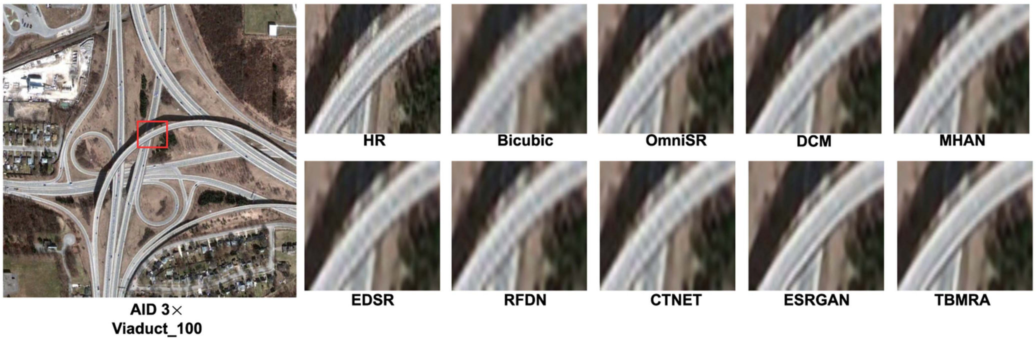


Fig. 3. Comparison of different models on the AID test data for 3 × SR.

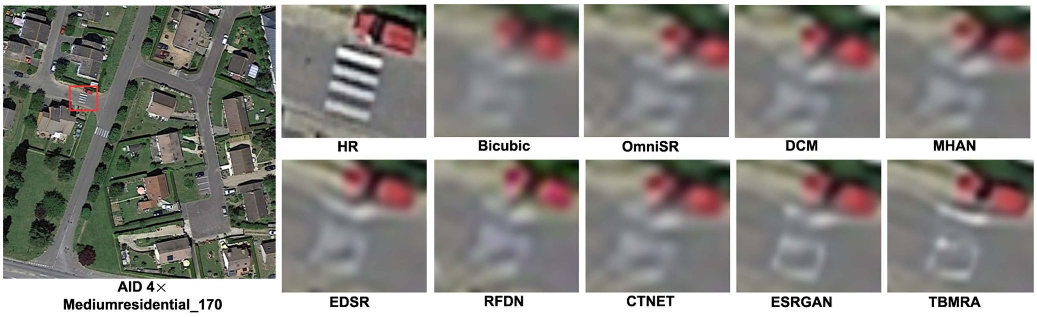


Fig. 4. Comparison of different models on the AID test data for 4× SR.

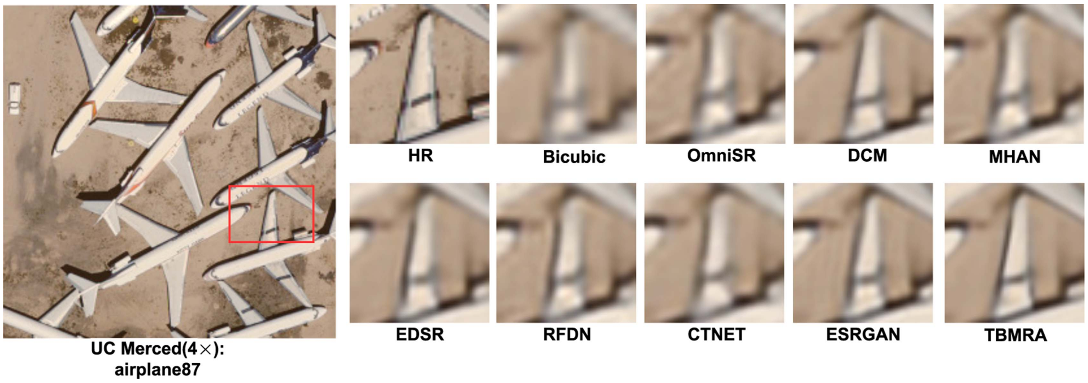


Fig. 5. Comparison of different models on the UC Merced test data for 4×SR.

All selected images were randomly cropped to a patch size of 144 × 144. To enhance generalization, robustness, and feature learning, we rotated, color jittered, and Gaussian blurred the LR images before training and added Gaussian noise to the HR images. The Adam optimizer trained our model for 200 epochs with $\beta_1 = 0.9$, $\beta_2 = 0.999$, and $\epsilon = 10^{-8}$. The learning rate $\eta = 1e^{-4}$ was halved at epoch 100. We used 20 residual groups with a batch size of 8. The training for all the compared models was maintained consistently with the same iteration to facilitate a fair evaluation. The network, implemented using the PyTorch library, was trained on an NVIDIA GeForce RTX 3080 GPU.

C. Evaluation Metrics

The results were validated using standard evaluation metrics, such as PSNR, SSIM, and learned perceptual image patch similarity (LPIPS) [44]. To estimate the PSNR index of the test image, we compared the super-resolved image with the original image. The PSNR can be calculated as

$$\text{PSNR} = 10\log_{10}\left(\frac{255^2}{\text{MSE}(I_x, I_y)}\right). \tag{30}$$

Mean squared error (MSE) can be defined as

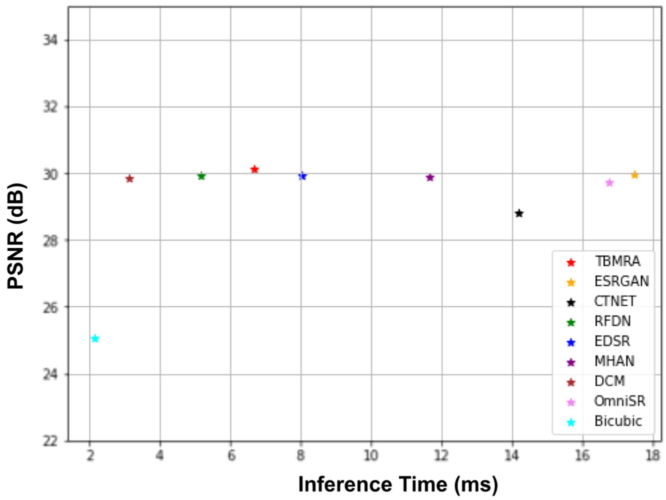


Fig. 6. PSNR versus inference time for the UC Merced dataset with scale factor × 4.

$$\mathrm{MSE} = \frac{1}{uv}\sum_{m=0}^{u-1}\sum_{n=0}^{v-1}(I_{x(m,n)} - I_{y(m,n)})^2 \tag{31}$$

where I_x denotes the input image, I_y denotes the super-resolution image, u and m denote the number of rows of pixels and index of the image, and v and n denote the number of columns of pixels and the index of that column. Higher PSNR values indicate better output.

The SSIM is calculated as

$$\mathrm{SSIM}(I_x, I_y) = \frac{(2I_{\mu_x}I_{\mu_y} + c_{s1})(2I_{\sigma xy} + c_{s2})}{(I_{\mu_x}^2 + I_{\mu_y}^2 + c_{s1}) + (I_{\sigma x}^2 + I_{\sigma y}^2 + c_{s2})} \tag{32}$$

where μ and σ denote the mean and standard deviation of a given image (*x*,*y*), and c_{s1} and c_{s2} denote constants for ensuring stability. A larger value for SSIM means the reconstructed output is closer to the original one.

The LPIPS metric is calculated as

$$I_{\mathrm{LPIPS}} = ||\phi(I_x) - \phi(I_y)|| \tag{33}$$

where ϕ is the pretrained AlexNet [45] network used for extracting features of the images. A lower value for LPIPS means a greater resemblance between two images.

D. Comparison With Existing State-of-the-Art Methods

We compared the proposed TBMRA model with these popular models: Bicubic, OmniSR [22], DCM [10], MHAN [15], EDSR [8] and RFDN [32], CTNET [46], and ESRGAN [47]. Tables I–III tabulate the quantitative results for the UC Merced, RSSCN7, and AID test datasets. Our model delivered the best restoration results for PSNR, SSIM, and LPIPS metrics at scaling factors of ×2, ×3, and ×4, as highlighted in bold font. Fig. 3 qualitatively compares TBMRA with state-of-the-art methods for the AID dataset's "viaduct100" image for a scale factor of ×3. TBMRA achieves superior performance by producing sharper edges and reconstructing more details of the rectangular object. Fig. 4 compares the "mediumresidential170" image with different methods for a scale factor of ×3. The result shows that TBMRA effectively reconstructs the marking and the red car. Fig. 5 shows a test image named "airplane87" from the UC Merced dataset with a scale factor of ×4. The TBMRA method reconstructs the airplane wings and other boundaries and edges more accurately than the other models. Thus, the proposed method obtained results closer to the HR references with sharper edges and textures.

TBMRA's strength lies in its attention modules and kernel sizes. A series of experiments were performed to evaluate their effectiveness for different channel and SA combinations. We also varied the kernel size to determine the combination of features resulting in high evaluation values. The results of these experiments are reported in Tables IV–VI.

Table IV compares the network with different CA models. We get the best result by combining ECA and SA. ECA is more effective at capturing long-range dependencies, and SA forces the network to focus on the essential regions. Table V compares the combination of different kernel sizes for effective feature extraction. For the combination of kernel sizes 3 and 5, we get better PSNR and SSIM values because the network can capture both fine-grained and larger-scale features. Table VI compares TBMRA with the state-of-the-art using 5, 10, 15, and 20 residual blocks, which correspond to "*N*" blocks in Fig. 1. The proposed model's PSNR and SSIM values were higher, given the same number of blocks. When we increased the number, the evaluation value increased as it extracted more features. For a fair evaluation, we used the same training skills and residual blocks to train all the models on the RSSCN7 dataset (as in Tables IV–VI). Fig. 6 compares PSNR versus inference time for the UC Merced dataset. Our TBMRA model's accuracy increased by fusing multiscale features without an unreasonably high inference time.

E. Super-Resolution on Real-World Images

In this section, we test our method on real-world satellite images. As there is no HR data, we evaluate the data with a nonreference image quality evaluator (NIQE) matrix [48] that extracts features from a multivariate Gaussian model. A lower value indicates better performance. Fig. 7 shows that our model can recover sharper and clearer edges closer to the ground truth with a lower value of the NIQE metric. This demonstrates our method's efficiency in real-world scenarios.

F. Color Error Map Investigation

We performed experiments to obtain our model's color error maps, as shown in Fig. 8. In the color maps, for the "g316" image corresponding to ×4 factor, the square and circle marks denoted the reconstructed fine-grain details. In the image "denseresidential89" for the upsample of ×2, only our model could restore the correct texture accurately, as indicated by the red rectangle. It also shows better recovery of fine object details than other models, as indicated by the circle in the image.

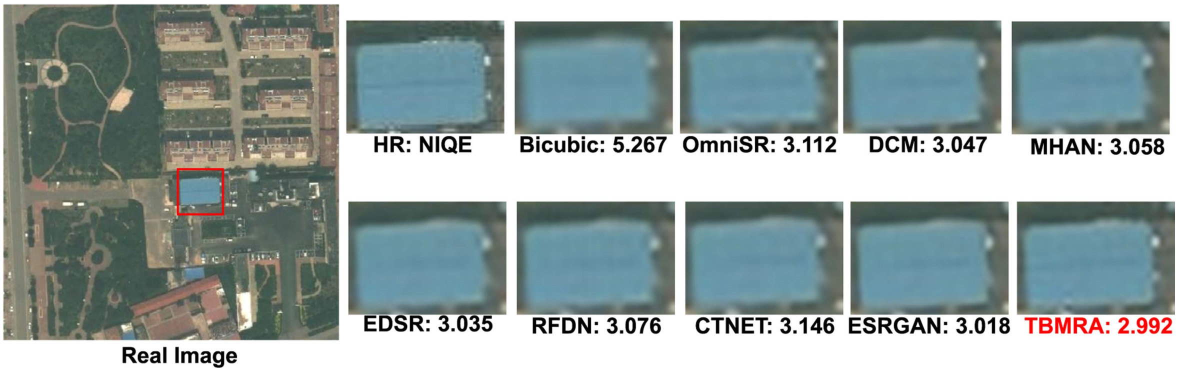


Fig. 7. Comparison for the Real-world Dataset. NIQE score is given under the reconstructed image.

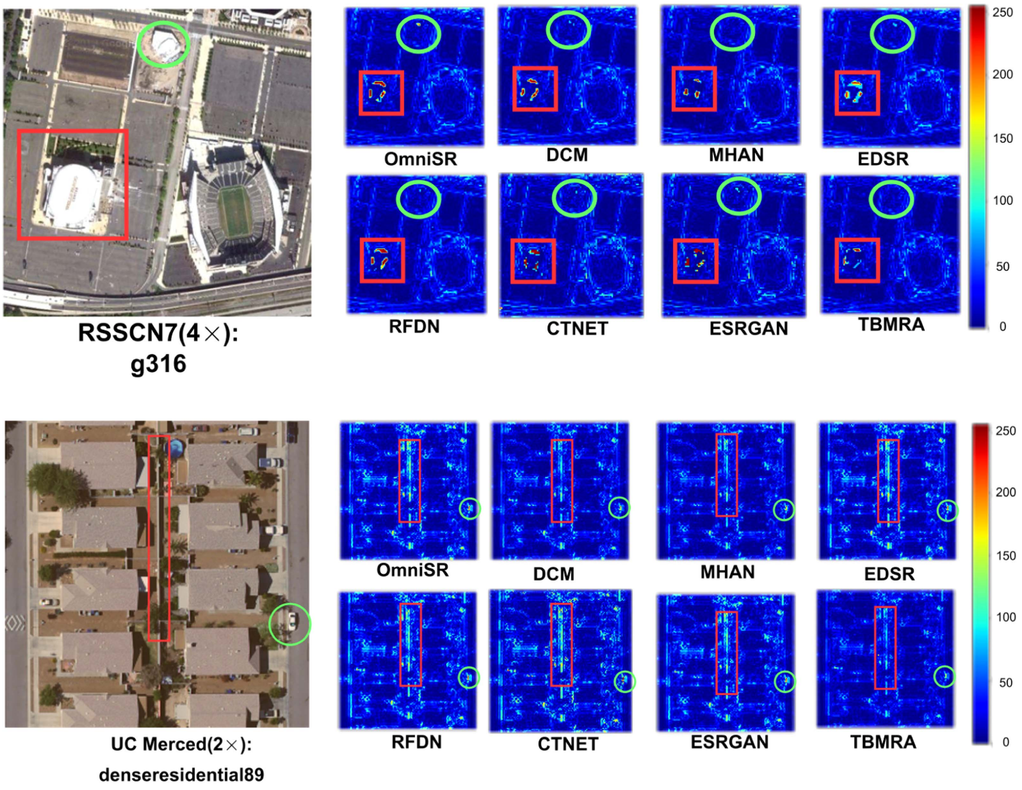


Fig. 8. Colored error maps to compare the different methods.

V. Conclusion

This article proposes a novel TBMRA CNN for super-resolution reconstruction of remote sensing images. The two branches are designed to extract features from the input image at different levels of detail. They are trained jointly to produce high-quality results. The proposed TBMRA network incorporates multiscale hybrid attention to extract low- and high-level image features and simultaneously emphasizes important features. To enhance model stability, the network is trained with local and global residual connections. Experimental results show our model's effectiveness in capturing and preserving salient information in images, surpassing other methods. Notably, TBMRA strikes a favorable balance between model performance and inference time among the models tested. Moreover, our approach demonstrated its practicality by showing promising results with real-world satellite imagery. We aim to extend our work to encompass multispectral imaging to acquire information at longer wavelengths.

References

[1] A. Joshi, B. Pradhan, S. Gite, and S. Chakraborty, "Remote-sensing data and deep-learning techniques in crop mapping and yield prediction: A systematic review," *Remote Sens.*, vol. 15, no. 8, 2023, Art. no. 2014.

[2] N. Farmonov et al., “Crop type classification by DESIS hyperspectral imagery and machine learning algorithms,” *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 16, pp. 1576–1588, 2023.

[3] F. A. Diaz-Gonzalez, J. Vuelvas, C. A. Correa, V. E. Vallejo, and D. Patino, “Machine learning and remote sensing techniques applied to estimate soil indicators–review,” *Ecological Indicators*, vol. 135, 2022, Art. no. 108517.

[4] S. Abu El-Magd, G. Soliman, M. Morsy, and S. Kharbish, “Environmental hazard assessment and monitoring for air pollution using machine learning and remote sensing,” *Int. J. Environ. Sci. Technol.*, vol. 20, no. 6, pp. 6103–6116, 2023.

[5] A. A. Khan, A. Jamil, D. Hussain, I. Ali, and A. A. Hameed, “Deep learning-based framework for monitoring of debris-covered glacier from remotely sensed images,” *Adv. Space Res.*, vol. 71, no. 7, pp. 2978–2989, 2023.

[6] 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. 2016.

[7] T. Tong, G. Li, X. Liu, and Q. Gao, “Image super-resolution using dense skip connections,” in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 4799–4807.

[8] B. Lim, S. Son, H. Kim, S. Nah, and K. Mu Lee, “Enhanced deep residual networks for single image super-resolution,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops*, 2017, pp. 136–144.

[9] Y. Zhang, Y. Tian, Y. Kong, B. Zhong, and Y. Fu, “Residual dense network for image super-resolution,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 2472–2481.

[10] J. M. Haut, M. E. Paoletti, R. Fernández-Beltran, J. Plaza, A. Plaza, and J. Li, “Remote sensing single-image superresolution based on a deep compendium model,” *IEEE Geosci. Remote Sens. Lett.*, vol. 16, no. 9, pp. 1432–1436, Sep. 2019.

[11] Z. Pan, W. Ma, J. Guo, and B. Lei, “Super-resolution of single remote sensing image based on residual dense backprojection networks,” *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 10, pp. 7918–7933, Oct. 2019.

[12] J. M. Haut, R. Fernandez-Beltran, M. E. Paoletti, J. Plaza, and A. Plaza, “Remote sensing image superresolution using deep residual channel attention,” *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 11, pp. 9277–9289, Nov. 2019.

[13] X. Dong, X. Sun, X. Jia, Z. Xi, L. Gao, and B. Zhang, “Remote sensing image super-resolution using novel dense-sampling networks,” *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 2, pp. 1618–1633, Feb. 2021.

[14] B. Niu et al., “Single image super-resolution via a holistic attention network,” in *Proc. Eur. Conf. Comput. Vis.*, Glasgow, U.K., Springer, 2020, pp. 191–207.

[15] D. Zhang, J. Shao, X. Li, and H. T. Shen, “Remote sensing image super-resolution via mixed high-order attention network,” *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 6, pp. 5183–5196, Jun. 2021.

[16] W.-Y. Hsu and P.-W. Jian, “Detail-enhanced wavelet residual network for single image super-resolution,” *IEEE Trans. Instrum. Meas.*, vol. 71, pp. 1–13, 2022.

[17] Z. Lu, J. Li, H. Liu, C. Huang, L. Zhang, and T. Zeng, “Transformer for single image super-resolution,” in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 457–466.

[18] F. Cao and H. Liu, “Single image super-resolution via multi-scale residual channel attention network,” *Neurocomputing*, vol. 358, pp. 424–436, 2019.

[19] P. Lei and C. Liu, “Inception residual attention network for remote sensing image super-resolution,” *Int. J. Remote Sens.*, vol. 41, no. 24, pp. 9565–9587, 2020.

[20] H. Huan et al., “End-to-end super-resolution for remote-sensing images using an improved multi-scale residual network,” *Remote Sens.*, vol. 13, no. 4, 2021, Art. no. 666.

[21] X. Zhang et al., “Remote sensing image super-resolution via dual-resolution network based on connected attention mechanism,” *IEEE Trans. Geosci. Remote Sens.*, vol. 60, pp. 1–13, 2021.

[22] H. Wang, X. Chen, B. Ni, Y. Liu, and J. Liu, “Omni aggregation networks for lightweight image super-resolution,” in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2023, pp. 22378–22387.

[23] Z. Wang, Y. Zhao, and J. Chen, “Multi-scale fast fourier transform based attention network for remote sensing image super-resolution,” *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 16, pp. 2720–2740, 2023.

[24] R. Dong, L. Mou, L. Zhang, H. Fu, and X. X. Zhu, “Real-world remote sensing image super-resolution via a practical degradation model and a kernel-aware network,” *ISPRS J. Photogrammetry Remote Sens.*, vol. 191, pp. 155–170, 2022.

[25] Z. Qiu, H. Shen, L. Yue, and G. Zheng, “Cross-sensor remote sensing imagery super-resolution via an edge-guided attention-based network,” *ISPRS J. Photogrammetry Remote Sens.*, vol. 199, pp. 226–241, 2023.

[26] Y. Xiao, Q. Yuan, K. Jiang, J. He, Y. Wang, and L. Zhang, “From degrade to upgrade: Learning a self-supervised degradation guided adaptive network for blind remote sensing image super-resolution,” *Inf. Fusion*, vol. 96, pp. 297–311, 2023.

[27] C. Szegedy et al., “Going deeper with convolutions,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 1–9.

[28] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 770–778.

[29] J. Li, F. Fang, K. Mei, and G. Zhang, “Multi-scale residual network for image super-resolution,” in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 517–532.

[30] Z. Tu et al., “AEFormer: Zoom camera enables remote sensing super-resolution via aligned and enhanced attention,” *Remote Sens.*, vol. 15, no. 22, 2023, Art. no. 5409.

[31] J. Sui, X. Ma, X. Zhang, and M.-O. Pun, “GCRDN: Global context-driven residual dense network for remote sensing image superresolution,” *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 16, pp. 4457–4468, 2023.

[32] J. Liu, J. Tang, and G. Wu, “Residual feature distillation network for lightweight image super-resolution,” in *Proc. Eur. Conf. Comput. Vis.*, Springer, 2020, pp. 41–55.

[33] S. Lei and Z. Shi, “Hybrid-scale self-similarity exploitation for remote sensing image super-resolution,” *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5401410,.

[34] Y. Song, J. Li, Z. Hu, and L. Cheng, “DBSAGAN: Dual branch split attention generative adversarial network for super-resolution reconstruction in remote sensing images,” *IEEE Geosci. Remote Sens. Lett.*, vol. 20, 2023, Art. no. 6004705.

[35] S. Jia, Z. Wang, Q. Li, X. Jia, and M. Xu, “Multiattention generative adversarial network for remote sensing image super-resolution,” *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5624715.

[36] W. Shi et al., “Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 1874–1883.

[37] Q. Wang, B. Wu, P. Zhu, P. Li, W. Zuo, and Q. Hu, “ ECA-Net: Efficient channel attention for deep convolutional neural networks,” in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 11534–11542.

[38] G.-S. Xia, W. Yang, J. Delon, Y. Gousseau, H. Sun, and H. Maître, “Structural high-resolution satellite image indexing,” in *Proc. ISPRS TC VII Symp.–100 Years ISPRS*, 2010, pp. 298–303.

[39] D. Dai and W. Yang, “Satellite image classification via two-layer sparse coding with biased image representation,” *IEEE Geosci. Remote Sens. Lett.*, vol. 8, no. 1, pp. 173–176, Jan. 2011.

[40] Y. Yang and S. Newsam, “Bag-of-visual-words and spatial extensions for land-use classification,” in *Proc. 18th SIGSPATIAL Int. Conf. Adv. Geographic Inf. Syst.*, 2010, pp. 270–279.

[41] Q. Zou, L. Ni, T. Zhang, and Q. Wang, “Deep learning based feature selection for remote sensing scene classification,” *IEEE Geosci. Remote Sens. Lett.*, vol. 12, no. 11, pp. 2321–2325, Nov. 2015.

[42] G.-S. Xia et al., “AID: A benchmark data set for performance evaluation of aerial scene classification,” *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 7, pp. 3965–3981, Jul. 2017.

[43] R. Dong, L. Zhang, and H. Fu, “RRSGAN: Reference-based super-resolution for remote sensing image,” *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2021, Art. no. 5601117.

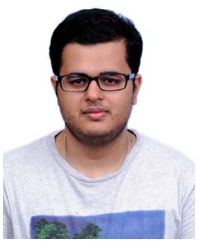
[44] R. Zhang, P. Isola, A. A. Efros, E. Shechtman, and O. Wang, “The unreasonable effectiveness of deep features as a perceptual metric,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 586–595.

[45] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “ImageNet classification with deep convolutional neural networks,” in *Proc. Adv. Neural Inf. Process. Syst.*, 2012, pp. 1106–1114.

[46] S. Wang, T. Zhou, Y. Lu, and H. Di, “Contextual transformation network for lightweight remote-sensing image super-resolution,” *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2021, Art. no. 5615313.

[47] X. Wang et al., “ESRGAN: Enhanced super-resolution generative adversarial networks,” in *Proc. Eur. Conf. Comput. Vis. Workshops*, 2018, pp. 63–79.

[48] A. Mittal, R. Soundararajan, and A. C. Bovik, “Making a ‘completely blind’ image quality analyzer,” *IEEE Signal Process. Lett.*, vol. 20, no. 3, pp. 209–212, Mar. 2013.



Allen Patnaik received the B.Tech degree in electronics and communication engineering from the G.I.F.T. Institute, Odisha, India, in 2014, and the M. Tech. degree in communication system engineering from the V.S.S.U.T University, Odisha, India, in 2018. He is currently working toward the Ph.D. degree in signal processing and machine learning with the EEE Deptartment, Indian Institute of Technology Guwahati, Guwahati, India.

His research interests include computer vision, deep learning, and remote sensing.

M. K. Bhuyan (Senior Member, IEEE) received the Ph.D. degree in EEE from the India Institute of Technology (IIT) Guwahati, Guwahati, India.

In the year 2014, he was a Visiting Professor with Indiana University, Bloomington, IN, USA, and Purdue University, West Lafayette, IN. He is currently a Professor with the Department of Electronics and Electrical Engineering, IIT Guwahati, and Dean of Infrastructure, Planning, and Management, IIT Guwahati. He is also currently working as a Visiting Professor, Department of Computer Science, Chubu University, Kasugai, Japan. He is also an Associate Faculty of the "Mehta Family School of Data Science and Artificial Intelligence" and "Centre for Linguistic Science and Technology," IIT Guwahati. He has almost 29 years of industry, teaching, and research experience. His current research interests include machine learning and artificial intelligence, and image/video processing.

Dr. Bhuyan was the recipient of the National Award for Best Applied Research/Technological Innovation, which was presented by the Honorable President of India in the year 2012, the Prestigious Fullbright-Nehru Academic and Professional Excellence Fellowship, and the BOYSCAST Fellowship.

Karl Fredric MacDorman received the Ph.D. degree in computer science from Cambridge University, Cambridge, U.K., in 1997.

He is an Associate Professor in the Human–Computer Interaction Program with the Luddy School of Informatics, Computing, and Engineering, Indiana University, Indianapolis, IN, USA. He is also the Director of the Informatics and Artificial Intelligence Programs and the Associate Dean of Academic Affairs. His research interests include cognitive science, human–computer interaction, machine learning, and robotics.