IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received October 16, 2020, accepted October 31, 2020, date of publication November 5, 2020, date of current version November 17, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3036155

A Convolutional Neural Network for Image Super-Resolution Using Internal Dataset

JING LIU[1], YUXIN XUE[1], SHANSHAN ZHAO[2], SHANCANG LI[1,2], (Member, IEEE), AND XIAOYAN ZHANG[3]

[1]School of Computer Science and Engineering, Xi'an University of Technology, Xi'an 710048, China
[2]Department of Computer Science and Creative Technologies, University of the West of England, Bristol BS16 QY, U.K.
[3]Information Science Technology College, Tan Kah Kee College, Xiamen University, Xiamen 361000, China

Corresponding author: Jing Liu (liujing@xaut.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 61472319, and in part by the Natural Science Basic Research Plan in Shaanxi Province of China under Grant 2019JM-467.

ABSTRACT Deep convolutional neural networks have recently achieved dramatic success in super-resolution (SR) performance in the past few years. However, the parameters of the mapping functions of these networks require an external dataset for training. In this paper, we propose a convolutional network for image super-resolution reconstruction that can be trained using an internal dataset constructed using a single image. The proposed single image convolutional neural network (SICNN) is designed with two branches. First, a large scale-feature branch trains the feature mappings that are from the low resolution (LR) image patches to the high- resolution image (HR) patches. The LR image patches are the enlarged image patches via bicubic interpolation. Second, the small scale-feature branch trains the feature mappings that are from the down-sampling image patches to the enlarged image patches. In contrast to the existing SR networks, the SICNN enjoys two desirable properties: 1) it does not require external datasets to conduct training, and 2) it enlarges an SR image at an arbitrary scale while restoring the clear edges and textures. The results of evaluations on a wide variety of images show that the proposed SICNN achieves advantages over the state-of-the-art methods in terms of both numerical results and visual quality.

INDEX TERMS Image super-resolution, convolutional neural network, internal dataset, arbitrary scale, enlargement.

I. INTRODUCTION

Super-resolution (SR) reconstruction seeks to restore high resolution (HR) images from one or more low resolution (LR) images. Super-resolution (SR) reconstruction demands that the image details can be still retained clearly while the reconstructed images are being zoomed in or out on. Super-resolution (SR) reconstruction is widely applied to video supervision, medical imaging, military reconnaissance, and remote surveillance [1]–[3]. This technology has therefore attracted great attention over the last three decades and many methods have been proposed. These methods can be divided into the following three categories: interpolation-based methods, reconstruction-based methods, and learning-based methods.

Interpolation-based SR methods [4], [5] estimate the missing pixel in an HR image using the neighborhood pixels in the input LR image. Although the complexity of interpolation-based methods is low, they usually result in blurred edges and image details. Therefore, this type of method can only be used as the preprocessing step of the SR task and not as an efficient scheme for high-quality SR reconstruction.

Reconstruction-based SR methods use the prior knowledge to reconstruct the HR image, that is, the reconstructed HR image should be very similar to the input LR image if the input is degraded by blurring and down-sampling. Reconstruction-based SR methods can be divided into the four subtypes: global prior-based [6], [7], local prior-based [8], [9], self-similarity prior-based [10], [11], and hybrid prior based [12] methods. The advantage of this kind of method is that they do not require extra external datasets while reconstructing the SR images. However, the performance of this kind of method is not desired when the enlarging scale factor becomes larger.

Learning-based SR methods [13]–[16] first learn the mapping relationship between LR and HR image pairs from an

The associate editor coordinating the review of this manuscript and approving it for publication was Md. Asikuzzaman.

external training dataset and then estimate its HR image from the test LR image using this mapping relationship. Learning-based SR methods can be classified into four subtypes: sparse coding-based [13], [14], regression-based [15], neighbor embedding-based [16], and deep learning-based methods [17]–[34]. Among these subtypes, deep learning methods based on a convolutional neural network (CNN) significantly improve the super-resolution reconstruction performance by learning many complex non-linear mappings.

The SRCNN [17] is the first convolutional neural network (CNN) to recover HR images from LR images, and it has made significant progress compared with traditional methods. Although SRCNN only has three convolution layers, its performance is very stable. The shortcoming of this network is that the input LR images must be the desired size, which will make the input LR images lose some details. The FSRCNN [18] uses LR images with arbitrary sizes as its input and exploits deconvolutional layers to enlarge the feature maps. Meanwhile, the proposed ESPCN [19] enlarges the feature maps by replacing deconvolutional layers with subpixel convolution layers. The subpixel convolution layer was adopted in both SRResNet [20] and EDSR [21] to magnify the final feature maps. However, the deconvolutional layer and the subpixel convolution layer methods can only magnify the feature maps at some certain integer scales (X2, X3, and X4). Meta-SR [22] proposed to replace the typical upscale module with a meta-upscale module that can dynamically predict the weight for each pixel, thus generating HR images with arbitrary sizes.

Some networks improved the image super-resolution reconstruction performance by enhancing the network depth. VDSR [23] increased the network depth and allowed the number of convolution layers to reach 20 by introducing residual learning. Many variants have been derived from VDSR, such as SRResNet [20] and EDSR [21], which contain more convolution layers than VDSR (SRResNet has 32 layers and EDSR has 64 layers). SRResNet also proposed propagating the feature maps of each layer into all subsequent layers to alleviate the vanishing gradient problem by using skip connections. The DRCN [24] first adopted a recursive learning to share parameters and increase the receptive field, thus improving the performance of the network. The DRRN [25] also introduced recursive blocks with shared parameters to make the network stable. The RDN [34] introduced direct connections among the layers within each dense block so that the network has the characteristic of dense connections. The dense connections further enhance the depth and the performance of the network.

Increasingly more CNN-based methods were devoted to learning the mapping function between the LR and HR image pairs to improve the quality of SR images. However, the reconstructed images from this type of method will easily generate artifacts when the difference between the input image and the training images is large because the mapping relationship of these methods heavily relies on the external dataset. Second, it is not just impossible but also unrealistic to provide an appropriate external training dataset for each image. ZSSR [26] filled the void and was the first and only CNN to train the feature mapping using the patches separated from a single image rather than an external dataset. The issue is that ZSSR only trained the mapping function of images in a single scale and failed to exploited the multi-scale of the images.

In this paper, we proposed a convolutional neural network not requiring an external training dataset for image SR reconstructing. The dataset used to train the parameters of the proposed convolutional neural network (SICNN) is composed of image patches separated from the test image itself. When the parameters of the SICNN tend to be stable, the test image that is zoomed in or out on is input into the SICNN, and the output of the SICNN will be the magnified SR image with an arbitrary scale. The general overview of the process for improving the performance of the SICNN is shown in Fig. 1. Fig.2 shows its architecture consisting of two branches: the large scale-feature branch is composed of convolution layers and its input is obtained from up-sampling on the HR image patches, and the small scale-feature branch is composed of convolution layers and deconvolution layers and its input is obtained from down-sampling on the HR image patches. Here, the HR image patches correspond to those separated from the test image. The purposes of using the two branches are the following: 1) image details can be clearly reconstructed by capturing image features at different scales, and 2) the network training time and the depth of the network can be reduced by expanding the width of the network. In addition, residual learning is still introduced into the large scale-feature branch, which ensures that the overall features are passed to the subsequent layers of the network. Experiments show that our proposed SICNN can reconstruct the HR images with arbitrary scale factors without an external training dataset.

The major contributions are summarized in three folds: 1) We propose a convolutional network (SICNN) to reconstruct HR images with an arbitrary scale factor using an internal dataset rather than an external training dataset; 2) We propose using dual branches to extract the image features from two scale image patches to improve the performance of the SICNN; and 3) We propose first enlarging the LR image to the proper size, then capturing the features and finally reconstructing its HR image.

II. RELATED WORK

Many strategies of upgrading the quality of SR images have been developed. Here, our discussions focus on the strategies that are relative to the proposed SICNN.

A. DATASETS

Most of the CNN-based SR reconstruction networks require thousands of images from an external dataset (for example, Set5 [27], Set14 [28], BSD100 [29], and Urban100 [30]) to be trained by learning a nonlinear mapping function between their HR and LR image pairs. In fact, it is external datasets

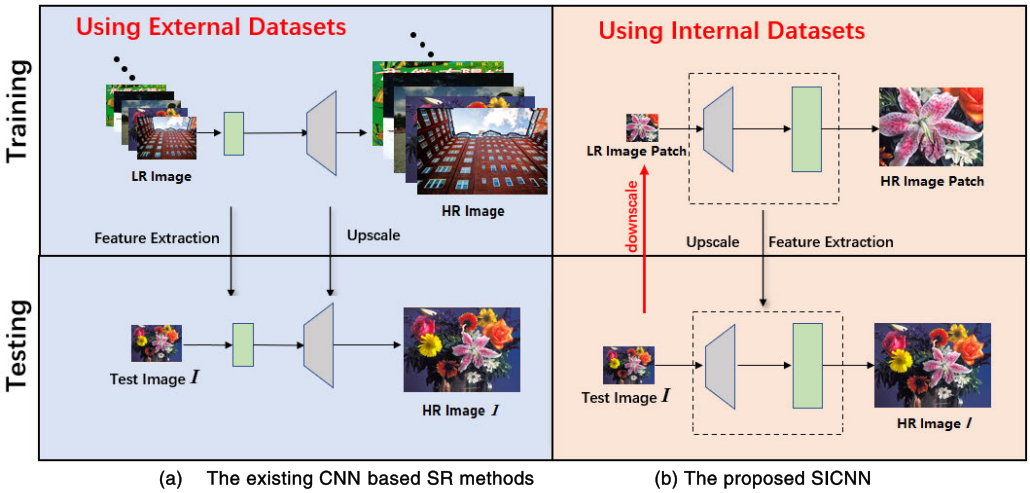


FIGURE 1. The comparison of the proposed SICNN and other CNN networks.

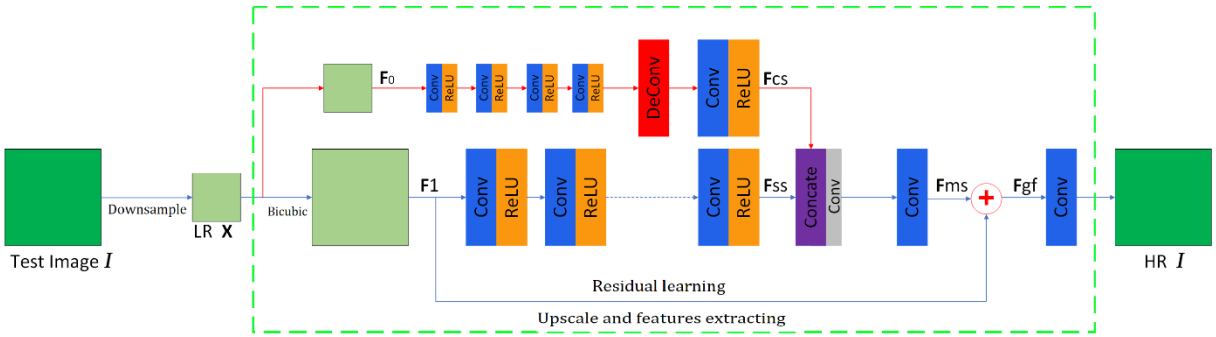


FIGURE 2. The architecture of the proposed SICNN.

that provide many training images so that CNNs can fully maximize their good modeling ability, thus significantly improving the image super-resolution reconstruction performance. However, the external datasets have some defects: first, in some applications such as hyperspectral images or SAR images reconstruction, we do not have enough available external datasets to train the CNN; second, although there are probably lots of images included in datasets, there is still no guarantee that each LR image can provide a suitable mapping relationship; and last, huge datasets need a complicated and time-consuming network to be competently used. The ZSSR network combined internal datasets with a CNN for the first time and used a large number of repeated LR image patches to train the feature mapping to HR image patches. Some experiments show that internal datasets have been proved to result in better experiment results when the degradation model of an HR image is unknown. Better quality SR images are more likely to be achieved when the appropriate external dataset cannot be obtained. Considering the advantages of internal datasets, this paper extends the multiscale features to the convolution neural network to fully mine the internal dataset and improve the network performance.

B. ARBITRARY ENLARGEMENT

As we all known, most existing SR networks based on a CNN only consider the super-resolution of some certain integer scale factors ($\times 2$, $\times 3$, and $\times 4$). These networks treat super-resolution of each scale factor (having not considered a non-integer factor) as an independent task and zoom in on the feature maps using deconvolution [18], [31], [32] or the sub-pixel convolution [19], [21], [33], [34]. The aforementioned networks must design a specific magnification scheme for each scale factor. Each magnification scheme can zoom in on the image only using a fixed integer scale factor. These issues limit the use of SR networks based on CNNs to real-world scenarios. Aiming at this problem, Hu *et al.* [22] proposed a meta-upscale module, which is composed of convolution layers, to magnify the feature maps. For an arbitrary scale factor, the coordinates of the pixels of the feature maps and scale-related vector are taken as the input of the meta-upscale module and the output is the weights of the corresponding pixels of the magnified images, thus generating HR images with arbitrary sizes by using these weights. We observe that the mechanism of meta method is the same as that of the bicubic interpolation method. The only difference is that the

weights of the former are trained via a convolutional neural network, and the weights of the latter are calculated using an interpolation formula. The dramatic success of the deep convolution neural network in super-resolution (SR) performance shows that the convolution neural network is better at capturing the abstract features of images. In view of the above reasons, the proposed SICNN directly trains each pixel value of the LR images to reconstruct the SR images with an arbitrary scale.

III. PROPOSED METHOD

As computing hardware is improved, CNNs get more powerful and take a longer time to train. This is because these networks are getting increasingly deeper, and the number of network parameters keeps growing. In fact, lots of convolutional layers or a complicated network structure are not necessary for the SR reconstruction of images with less details and many smooth areas. In addition, the dataset constructed from a single image itself can meet the requirement of training the feature mapping relations of these images. This motivates us to design a small CNN to reconstruct an SR image without an available external dataset. The proposed SICNN is designed with the two branches and fulfils four functions: small scale-feature mapping, large scale-feature mapping, multi-scale feature fusion, and residual learning. The architecture of the proposed SICNN is shown in Fig.2. Before we detail the proposed the SICNN, the generation of the internal dataset is first introduced.

A. INTERNAL DATASET GENERATING

The SICNN is designed to be trained with an internal dataset. Compared with an external dataset, an internal dataset has better adaptability and a shorter training time. The tested image is used to create the internal dataset. This internal dataset is further augmented through flipping, rotating, and translation those image patches separated from the test image. The size of the image patch of the internal dataset depends on the enlargement factor. Supposing that the size of the test image is N and the enlargement factor is r, there can be up to $N \times N$ image patches in the constructed internal dataset and the size of each patch should be $N \times r/2$. The input of the small scale-feature branch is the LR image patches (denoted as F_0 in Fig.2) which are obtained via down-sampling those image patches belonging to the constructed internal dataset, and whose size is $N \times r/4$. The input of the large scale-feature branch is the LR image patches (denoted as F_1) which are obtained via bicubic interpolation those image patches belonging to the constructed internal dataset, and whose size is $N \times r/2$. The output of the SICNN should be an image with the size of $N \times r/2$. In the training stage of the network, the image patches in the dataset are considered as HR image patches, and the LR image patches (denoted as X in Fig.2, and their size is same as the size of the LR image patch F_0) are obtained via down-sampling the HR image patches. The proposed SICNN is trained by learning the mapping functions between HR and LR image patches. In the reconstructing stage of an image, the input of the SICNN should also be the test image with a size of $N \times r/2$.

B. LARGE-SCALE FEATURE MAPPING

There are generally two methods to learn the mapping between the LR image patches and the HR image patches. The first one is to enlarge the LR image to the proper size via interpolation and then extract the features from the interpolate LR image using convolution kernels, which might result in some detail features of the image being lost. The second method is to directly extract the features from the LR image using convolution kernels and then enlarge the image to the output size using deconvolution at the end of the network. Although the second method can greatly shorten the training time, the detail features with the same scale might be lost. Neither of the two methods can train fully the feature mapping of the network. Considering the advantages of multiscale features, the SICNN uses two branches, the feature mapping of the small and the larger scales, to extract the features at different scales from the LR image. The bottom branch shown in Fig.2, consisting of 8 convolution layers followed by 8 rectified linear units, corresponds to the large scale-feature mapping of the SICNN. The input is the LR image patches (denoted as F_1) obtained from bicubic interpolation, and the output is F_{ss}, which can be represented as

$$F_{ss} = H_{ss}(F_8) \tag{1}$$

where H_{ss} is a composite function including three consecutive operations: a 3×3 standard convolution, batch normalization (BN), and rectified linear units (ReLU); F_8 denotes the feature-maps generated by the 8^{th} convolution layer, totaling 64 feature-maps. The number of feature maps of each convolution layer in the large scale-feature branch is 64.

C. SMALL-SCALE FEATURE MAPPING

Large scale-feature mapping may not be able to learn some small feature details because the bicubic interpolation will cause these details to be too smooth to capture. To correct the issue caused by bicubic interpolation, we also adopt small scale-feature mapping to capture the small feature details directly from the LR image patches and then use deconvolution to enlarge the feature maps to the required size.

The deconvolution layer amplifies the feature maps with a set of deconvolution filters, and it can be thought of as the inverse operation of convolution. Just as convolution is used to extract features by means of a convolution kernel, the deconvolution can also capture the feature details. In addition, the deconvolution operation can enlarge the features to the desired resolution of the output image by padding zeros between the pixels of the LR image patch and then convolving the padding image to the desired resolution. Moreover, the deconvolution can simplify the calculation and speed up the convergence of the loss function.

In the small scale-feature mapping branch, we extract the features from the input LR image X directly using 4 convolution layers followed by rectified linear units (ReLU),

and then increase the resolution and the output size through 2 deconvolution layers. The output of the small scale-feature mapping branch is denoted as F_{cs}, which is concatenated with the large scale-feature maps into the multiscale features.

D. MULTI-SCALE FEATURE FUSION AND RESIDUAL LEARNING

The small scale-feature mapping and the large scale-feature mapping produce two sets of 64 channel feature maps containing the features extracted from the image patches with different scales. Multiscale feature fusion concatenates these two sets of feature maps and gets a set of 64 channel feature maps again, which can be denoted as

$$F_{ms} = H_{ms}([F_{ss}, F_{cs}]) \tag{2}$$

where $[F_{ss}, F_{cs}]$ denote the concatenation of the convolution layers of the small scale-feature mapping and the large scale-feature mapping, respectively, and H_{ms} refers to a composite operation of 1×1 and 3×3 convolutions. First, 1×1 convolution kernel is used to fuse the multi scale features and reduce the number of the feature maps, and then a 3×3 convolution kernel is used to extract the multiscale features again and realize residual learning.

Although a small CNN meets the requirement for reconstructing SR images with fewer details and many smooth areas, it is crucial to ensure that the overall features of images are passed from the current convolution layer to the next. To end this, we introduced residual learning [31] to directly connect the overall features of the LR image patches to the subsequent convolution layer. The final output of the SICNN is denoted as

$$F_{gf} = F_{ms} + F_1 \tag{3}$$

IV. EXPERIMENTS

The motivation of the SICNN that was designed is how to reconstruct HR images using a deep convolutional neural network when we have no external dataset suitable for training at hand. We conduct several experiments to validate the performance of the SICNN. First, we examined the SR images from the SICNN that was respectively trained with an internal dataset and an external dataset. Second, we examined the effect of multiscale feature mapping on the SICNN. Third, we examined the magnification performance of the SICNN. These three experiments are our ablation study, displaying the impact of the basic components on the performance of the SICNN. Finally, we selected six representative SR methods as the comparison baselines to evaluate the SR image quality of the SICNN: bicubic interpolation; a reconstruction-based SR method that requires no external datasets (CTV-DNLM [11]),three CNN-based methods that require external datasets (SRCNN [17], the RDN [34] and the meta network [22]), and the ZSSR network [26] that uses internal datasets.

The peak signal-to-noise ratio (PSNR) and structural similarity (SSIM) [35] are used to evaluate the SR results of the simulation images. We also use the information fidelity criterion (IFC) to measure the restored image details. Ref. [36] pointed out that the SSIM index and the IFC index are better for evaluating the fidelity of detail features. The higher the values are, the more similar the local structures of the image will be and the better the fidelity of the image will be. For the images from real-world scenarios, we introduced reference-free image quality evaluation metrics such as the NIQE [37] and SSEQ [38] to evaluate the SR quality due to the lack of the corresponding true HR images. These two indices are sensitive to the detail contrast, texture diversity and content sharpness of the images without the requirement for HR reference images. Therefore, they are particularly suitable for the quality evaluation of images from real-world scenarios [39]. The smaller the NIQE and the larger the SSEQ, the better the image quality.

A. EXPERIMENTAL SETTING

In the training phase, many patches are separated from the tested image and are down-sampled to get the LR image patches. The LR image patch X via down-sampling operation is taken as the input of the SICNN and the original image patch is taken as the ground truth image patch or the HR image patch. Supposing that the test image has the size of $N \times N$, there can be up to $N \times N$ image patches in the internal training dataset. This internal dataset is further augmented through applying the flipping, rotating, and translation operations to those image patches separated from the test image. The size of the training patches is determined according to the scale factor. If the size of the test image is 100×100 ($N = 100$) and the scale factor is 1.2, the number of the training patches will be 100×100 and each patch should have the size of 60×60. Meanwhile, the LR image patch X from the down-sampling operation will be 30×30. Once the network is trained, the test image (whose size should be 60×60 via down-sampling operation) can be fed into the network. The output will be the desired high-resolution image (whose size will be 120×120) and be enlarged by 1.2 times.

In the SICNN, except for the 1×1 convolutional layer after the concatenation, the other convolutional layers consist of 64 filters with a size of 3×3. We use the L_1 smooth loss with the ADAM optimizer and start with a learning rate of 0.001. The final convolution layer is 3 or 1, where the former corresponds to a color image and the later refers to a gray image.

B. ABLATION STUDIES

1) THE EVALUATION OF THE INTERNAL DATASET

In this section, we aim to verify that the SR qualities of some images can also be obtained by learning their own characteristics through the proposed SICNN when an external dataset is not available. To end this, we first respectively construct internal datasets using two different types of images. If SICNN is trained by using the internal dataset that is constructed by the image (as shown in Fig.3(b)) who has a

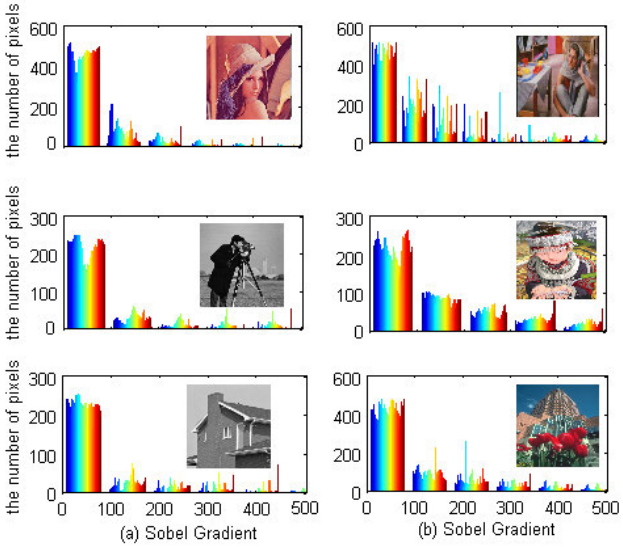


FIGURE 3. The distributions of the Sobel gradient of the images used to construct 6 internal datasets.

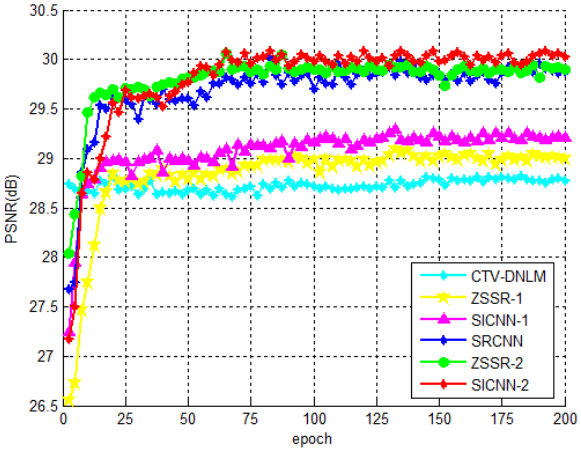


FIGURE 4. The comparison of PSNRs of using the internal dataset and the external dataset.

more decentralized distribution of gradient values, SICNN is represented as SICNN-1. If SICNN is trained by using the internal dataset that is constructed by the image (as shown in Fig.3(a)) whose gradient distribution is relatively concentrated within the small gradient value, SICNN is denoted as SICNN-2.The abscissa of Fig. 3 represents the mean square value of Sobel gradient and its ordinate indicates the number of pixels with the same Sobel gradient. Each color represents a bin that reveals the underlying distribution of the number of pixels within a certain range of Sobel gradient. There are relatively few pixels with large gradient values in the images shown in Fig. 3(a), There are relatively a lot more pixels with different gradient values in the images shown in Fig. 3(b). We took the SRCNN [17], the ZSSR [26] and the CTV-DNLM [11] as the comparison baselines. Regarding these methods, the SRCNN requires an external dataset for training, and 10 thousand images of its external dataset came from ISLVRC 2012 [40] and the Waterloo Exploration Database [41]; the ZSSR network and CTV-DNLM method based on reconstruction are the same as our proposed SICNN and do not need an external dataset. An image with a size of 256*256 can construct an internal dataset containing 65536 image patches. This number of patches is equivalent to the number of images in an external dataset training a network. The PSNRs obtained from these methods are shown in Fig.4. In Fig.4, ZSSR is represented as ZSSR-1 when it is trained by using the same dataset with SICNN-1, else ZSSR is denoted as ZSSR-2.

It can be seen from Fig.4 that when the performance of the networks tends to be stable, the PSNRs obtained from SICNN-2 are higher than those of the other methods. The results from SICNN-1 are lower than those of SRCNN but higher than those of the ZSSR-1 network and CTV-DNLM method. This demonstrates that the proposed SICNN is good at reconstructing the SR images containing not too much different gradient information via the internal dataset. For those images with a relatively centralized distribution of gradient values, the self-similarity of an image is very high. The SICNN-2, which is trained using the internal dataset constructed from highly similar images, is bound to be very familiar with this image, and so it can acquire better SR images. For those images with a more decentralized distribution of gradient values, the performance of the SICNN-1 is also superior to those of the ZSSR network and CTV-DNLM method that do not require an external dataset. The PSNR numerical results shown in Fig.3 illustrate that although the overall performance of the proposed SICNN is not as good as that of the SRCNN using the external dataset, it offers significant advantages over other SR methods using the external dataset when the external dataset is not available.

2) THE BEHAVIOR OF THE MULTI-SCALE FEAURE MAPPING

The input image is enlarged to the proper size via interpolation and then the features are extracted from the interpolated LR image using convolution kernels, which might lead to some detail features of the input image being lost. To make up for its weakness, we also directly extract the features from the input image using convolution kernels and then enlarge the feature maps to the output size via deconvolution. In the SICNN, we designed the small scale and the larger scale feature mapping branches to respectively extract the features at different scales from the input image. To verify the effectiveness of this scheme, we conducted the following experiments: the feature extracting and the feature learning are operated just using the large scale-feature mapping branch and the small scale-feature mapping branch is omitted. We denoted such network as SICNN-1. Like the last experiment, we took the ZSSR network as the comparison baseline method because it is also a super-resolution reconstruction algorithm for an internal dataset and examined the PSNRs, SSIMs and IFCs obtained from the ZSSR network, SICNN-1 method and SICNN method.

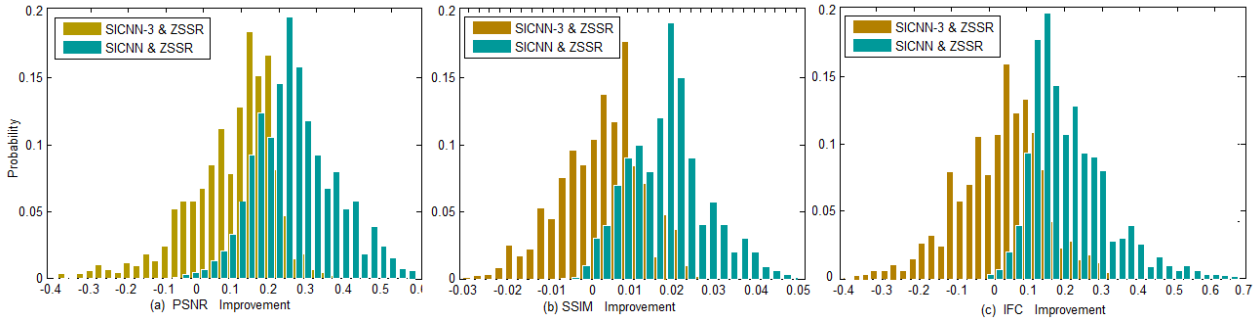


FIGURE 5. Probability distributions of PSNR, SSIM and IFC gains for 50-images with enlargement scale of 2.5.

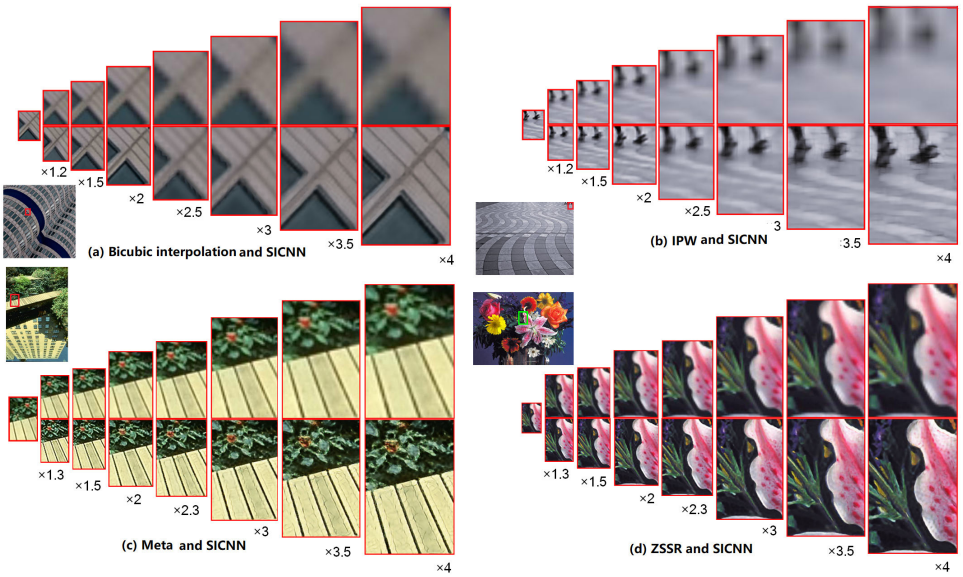


FIGURE 6. The visual comparison of the magnification with arbitrary scales, the bottom row of each subgraph is the result of the SICNN.

Fig. 5 shows the probability distributions of the PSNR, SSIM and IFC gains of the 50 SR images with a magnification factor of 2.5. These gain values are respectively obtained from the proposed SICNN and SICNN -1 relative to the baseline ZSSR method. The statistical results illustrate that the performance of the SICNN is significantly superior to that of the ZSSR method. Meanwhile, the performance of SICNN-1 is very close to that of the ZSSR method. This indicates that the strategy that adopts the small scale and the larger scale feature mapping branches to respectively extract the features at different scales for the LR image can significantly improve the SR performance of the SICNN.

3) THE MAGNIFICATION WITH ARBITRARY SCALE

There are not many super-resolution algorithms that focus on any scale factor. Thus, we use the following methods as baselines to evaluate the magnification superiority of the SICNN. Bicubic interpolation can be used to enlarge images. Therefore, the first baseline is the bicubic interpolation method used to enlarge the HR image. The second baseline (IPW) interpolates the HR image to the needed size and then uses the convolutional layers to predict the weights for each interpolation pixel. The third baseline is the meta method [22] which designs a special convolutional module to predict the weights of each pixel and utilizes the residual dense block (RDB) [34] to capture and train the feature of the input image. The fourth baseline is the ZSSR network [26] which only uses a single branch of small scale-features to capture and train the features from the input image. We train all these models on an arbitrary scale factor together.

The experimental results are shown in Fig.6. The top row of each subgraph is the visual result of the baseline method and the bottom row is the visual result of the SICNN. The top row of Fig. 6 (a) shows that the stripes obtained from the bicubic interpolation are too blurry to distinguish. This shows that upscaling the input image with only bicubic interpolation not only cannot make any texture or details clear, but it also causes many details to be lost. Both the IPW method and the meta method are used to learn the optimal weights of the interpolated pixel for an arbitrary scale factor via

TABLE 1. The average results of PSNR (dB), SSIM, and IFC from the BI degraded images for arbitrary scale factors.

Dataset	Scale	Methods						
		Bicubic	RDN	CTV-DNLM	SRCNN	ZSSR	Meta	SICNN
Set5	× 2	27.96	33.27	31.18	33.01	33.84	**33.85**	33.76
		0.7123	0.8875	0.8302	0.8530	0.8865	0.8938	**0.9017**
		1.5495	2.3844	1.8289	2.0097	2.2241	2.4297	**2.4721**
	× 3	24.04	30.46	29.16	30.64	30.93	30.89	**31.25**
		0.5697	0.7758	0.7467	0.7772	0. 7833	0. 7782	**0.7869**
		1.4382	1.5917	1.5382	1.5765	1.5978	1.5973	**1.5988**
	× 1.2	26.48	-	30.23	-	30.75	30.64	**31.08**
		0.6621	-	0.7443	-	0.7905	0.7841	**0.8021**
		1.5047	-	1.6336	-	1.6558	1.6532	**1.6605**
Set14	× 2	26.01	33.04	30.65	31.94	32.15	**33.24**	33.23
		0.6834	0.8479	0.8068	0.8127	0.8466	0.8475	**0.8524**
		1.5358	**2.2354**	1.7125	1.9276	2.1728	2.2263	2.2212
	× 3	23.04	30.08	28.79	30.42	30.75	30.82	**30.97**
		0.5385	0.7753	0.7531	0.7654	0.7693	0.7712	**0.7792**
		1.4571	1.5694	1.5496	1.5499	1.5674	1.5712	**1.5869**
	× 1.2	24.63	-	27.38	-	30.16	30.29	**30.33**
		0.5465	-	0.7803	-	0.7938	0.8047	**0.8119**
		1.4892	-	1.6337	-	1.6453	1.6351	**1.6467**
BSD100	× 2	25.17	**33.48**	30.38	32.09	33.43	33.42	33.21
		0.6929	0.8276	0.8067	0.8293	**0.8327**	0.8324	0.8309
		1.5021	2.2353	1.8255	1.9547	**2.1369**	2.2357	2.2164
	× 3	23.63	31.99	29.06	31.36	31.97	32.23	**32.62**
		0.5899	0.7812	0.7612	0.7724	0.7812	0.7873	**0.7902**
		1.4418	1.6049	1.5525	1.5671	1.5836	1.6114	**1.6256**
	× 1.2	24.36	-	29.35	-	32.73	32.64	**32.81**
		0.6127	-	0.7879	-	0.7995	0.7907	**0.8073**
		1.4863	-	1.5648	-	1.6476	1.6324	**1.6792**
Urban100	× 2	26.24	32.06	29. 83	31.92	**32.83**	32.72	32.65
		0.6392	0.8294	0.7674	0.8135	0.8279	**0.8306**	0.8301
		1.5352	2.0245	1.8556	1.9306	2.0344	2.0121	**2.1437**
	× 3	24.15	31.47	28.74	29.48	30.85	31.06	**31.87**
		0.5993	0.7994	0.7023	0.7551	0.7921	0.8071	**0.8133**
		1.4418	1.6125	1.5546	1.5789	1.5936	1.6086	**1.6354**
	× 1.2	25.34	-	29.63	-	**32.61**	30.97	32.16
		0. 6127	-	0.7279	-	0.8192	0.7892	**0.8235**
		1.4736	-	1.5638	-	1.6865	1.6565	**1.6920**
Manga109	× 2	25.36	32.53	29.27	31.13	32.41	32.42	**32.91**
		0.6185	0.8126	0.7463	0.7927	0.8126	**0.8185**	0.8164
		1.5129	1.9507	1.8214	1.9042	1.9695	1.9657	**2.0456**
	× 3	22.91	29.92	27.98	29.12	30.19	31.04	**31.98**
		0.5727	0.7823	0.7021	0.7342	0.7687	**0.8074**	0.7902
		1.4344	1.5987	1.5468	1.5669	1.5812	1.6093	**1.6256**
	× 1.2	23.16	-	28.35	-	31.88	31.54	**32.03**
		0.6038	-	0.7196	-	0.8098	0.8092	**0.8176**
		1.4563	-	1.5573	-	1.6254	1.6476	**1.6827**

convolutional layers. The difference is that the former uses the same convolution kernel for the different scale factors while the latter uses different convolution kernels. Convolution is good at capturing and learning the abstract features of an image but not good at computing the weights of pixels. The ZSSR network and the proposed SICNN are to capture the features of the enlarged images using the convolutional layers. The difference is that the SICNN not only captured the features from the enlarged image, but it also captured the features from the original image patches. Therefore, although the visual results from the meta method and the ZSSR method are both better than those of the IPW method, they are still clearly inferior to those of the proposed SICNN (shown in the bottom row of each subgraph). As the magnification increases, the shoe heels from the IPW method (Fig.6(b)) are not quite clear; the strips from the meta method (Fig.6(c)) are somewhat unclear, and the petals from the ZSSR method (Fig.6(d)) are somewhat clear. However, the visual results of these images obtained from the SICNN are considerably clearer. This demonstrates that the proposed SICNN has good magnification performance and is more efficient than these baseline methods.

C. COMPARISON WITH OTHER METHODS

To comprehensively verify the performance of the proposed SICNN, we examine the SR results of the proposed SICNN on simulated images and real-world images, respectively. The simulated images are the images degraded by applying certain processes on some images from the datasets, including the BI degraded images and the DN degraded images. We processed some images selected from the five datasets (Set5 [27], Set14 [28], BSD100 [29], Manga109, and Urban100 [30]) in the same way as Ref. [34]. Using the MATLAB environment,

FIGURE 7. Visual comparison of SR results (×2 scale) of BI degraded image *comic*.

BI degraded images are obtained by using bicubic downsampling, and DN degraded images are obtained by adding Gaussian noise at a level of 10. Meanwhile, the scale factor is set to 2 times, 3 times, and 1.2 times, respectively, to examine the magnifying performance of the SICNN. The regions of interest (ROIs) in each resultant image are magnified using bicubic interpolation with a scale factor of 2 or 3 and shown in the corners to compare the high frequency details obtained from different methods.

1) COMPARISONS ON BI DEGRADED IMAGES

The quantitative assessment values in terms of the PSNR, SSIM, and IFC obtained from the SRCNN [17], the meta network [22], the RDN [34], bicubic interpolation, the CTV-DNLM method [11], and the ZSSR network [26] are presented in table 1. The SICNN achieves the better metric values. In particularly, the three metrics of the SICNN network outperform those of the SRCN, meta network, and the RDN by large margins when some relatively smooth images are magnified with a non-integer power of 2. For the SRCNN, the RDN, and meta network that require external datasets, when suitable training datasets are available, these deep convolutional neural networks can have powerful learning capabilities by integrating multiple convolutional layers and providing accurate predictive features for the SR image. However, these networks all first reconstruct the LR images and then the obtained HR images are enlarged using other approaches. That is, reconstruction and magnification are two separate tasks, which may degrade the quality of the obtained HR images. In the SICNN, an LR image of any arbitrary size and ratio can be its input, which means that the enlarged image can also be used as the input of our network. When the image is enlarged, the details of the image may be lost. To overcome this problem, we designed a double branch of convolution layers: the bottom branch trains the large scale-feature mappings, and the top branch trains the small scale-feature mappings and enlarges the obtained feature maps via deconvolution. These components are useful for improving the numerical indicators of the SICNN. This is also confirmed by the visual results of the HR images.

The visual quality of BI degraded images (comic and imag_068) are shown in Fig. 7 and Fig.8. Bicubic interpolation has the worst visual quality despite its very easy and fast operations. Owing to being able to extract multidirectional and anisotropic features, the CTV-DNLM method can

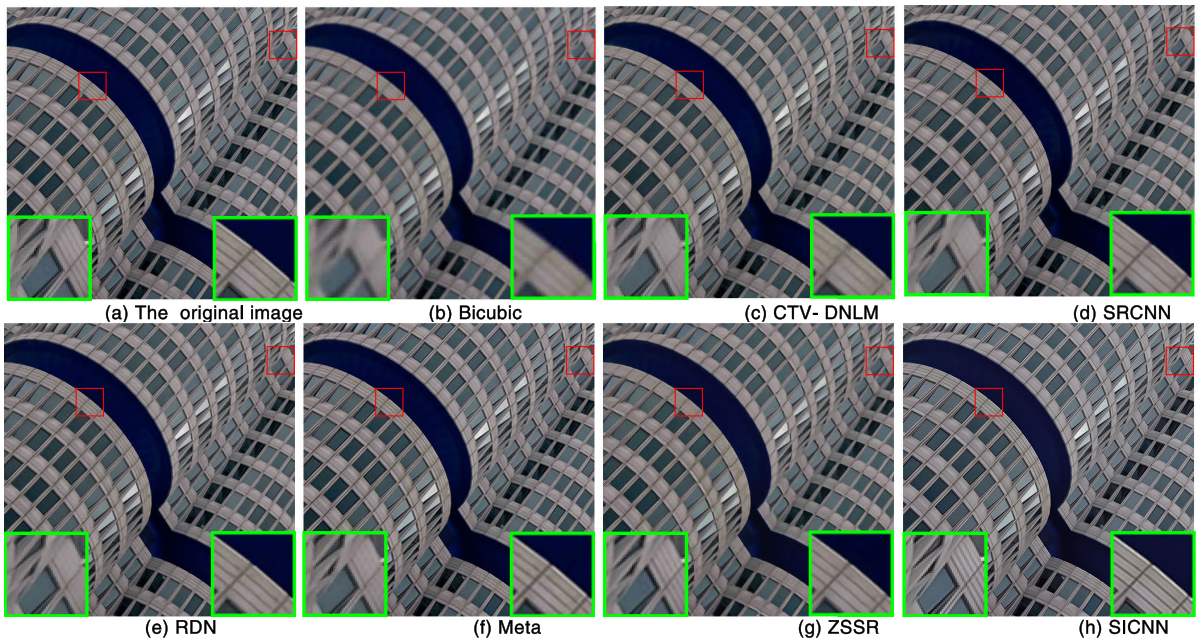


FIGURE 8. Visual comparison of SR results (×3 scale) of BI degraded image *img_068*.

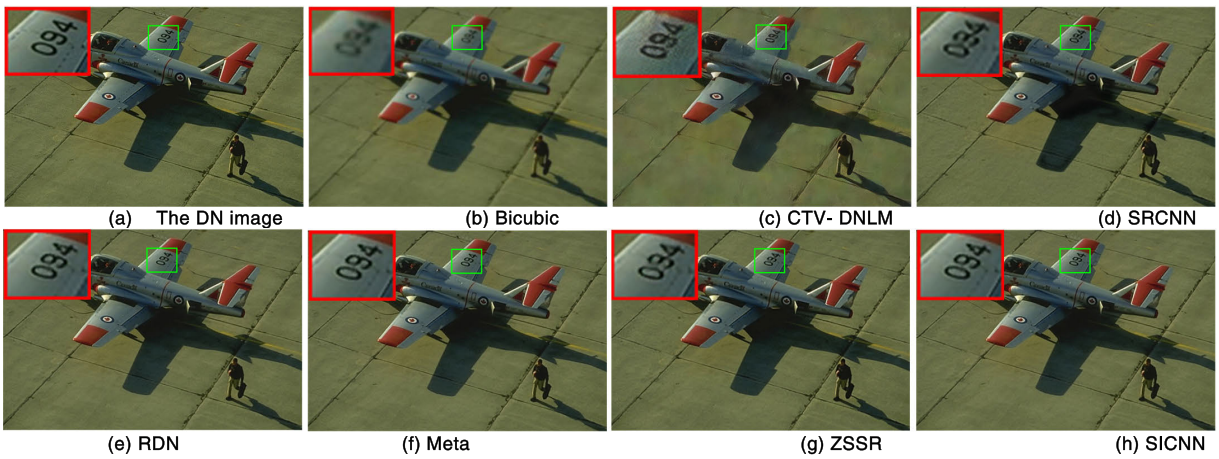


FIGURE 9. Visual comparison of SR results (×3 scale) of DN degraded image *img_071*.

estimate more detail information from the images compared with bicubic interpolation method, but some obvious artifacts also appear along the edges. The SRCNN, the RDN, and the meta network are the three deep convolutional neural methods using external training datasets and they achieved satisfactory visual results of the SR images. The meta network and the RDN are good at restoring the edge and texture details and only produces slight artifacts at the sharp edges. The RDN method [34] is far better than the CTV-DNLM method. However, compared with the SCICNN, the RDN loses quite many details along the edges and the edges look very unnatural. The meta network is clearly inferior to our SICNN both in reproducing natural-looking details and preserving sharp edges when the magnification images with a non-integer power of 2 must be reconstructed. This may result from the difference between computing the weights of the interpolation pixels and capturing the features of the interpolation pixels. The SICNN achieved SR images with better visual quality than those of the meta network and the RDN. Although the ZSSR network is also a convolutional neural method using an internal training dataset and can also be applied to SR images with any size, the captured and trained features are from only the enlarged images and the enlarged images have already lost considerable details, thus resulting in noticeable jagged artifacts being produced in the textured regions. By comparison, the SICNN was trained not only using the large scale-feature map but also using the small scale-feature map; therefore, it can capture and train as

TABLE 2. The average results of PSNR (dB), SSIM, and IFC from the DN degraded images for arbitrary scale factors.

Dataset	Scale	Methods						
		Bicubic	RDN	CTV-DNLM	SRCNN	ZSSR	Meta	SICNN
Set5	× 2	26.85	32.68	30.07	30.90	32.73	**32.74**	32.65
		0.7012	0.8793	0.8291	0.8419	0.8754	**0.8827**	0.8806
		1.5384	2.3089	1.8178	1.9986	2.2130	2.3186	**2.3610**
	× 3	23.93	29.47	28.05	29.53	29.82	29.78	**30.14**
		0.6586	0.7685	0.7356	0.7661	0. 7722	0. 7671	**0.7758**
		1.4271	1.5852	1.5273	1.5862	1.5867	1.5864	**1.5857**
	× 1.2	25.37	-	29.12	-	29.64	29.53	**29.97**
		0.6512	-	0.7332	-	0.7795	0.7730	**0.7913**
		1.4936	-	1.6225	-	1.6447	1.6421	**1.6496**
Set14	× 2	25.49	31.99	29.54	30.83	31.04	**32.13**	32.12
		0.6723	0.8374	0.7957	0.8016	0.8355	0.8364	**0.8413**
		1.5247	2.1246	1.7014	1.9165	2.1617	**2.2152**	2.2103
	× 3	21.93	30.57	27.68	29.31	30.75	30.82	**30.97**
		0.5274	0.7601	0.7423	0.7543	0.7582	**0.7641**	0.7632
		1.4462	1.5628	1.5385	1.5387	1.5563	1.5601	**1.5758**
	× 1.2	23.52	-	26.27	-	29.05	29.18	**29.22**
		0.5354	-	0.7692	-	0.7827	0.7936	**0.8008**
		1.4781	-	1.6226	-	1.6342	1.6243	**1.6356**
BSD100	× 2	24.06	**32.39**	29.27	30.91	32.22	32.32	32.10
		0.6818	0.8134	0.7956	0.8182	0.8216	**0.8217**	0.8191
		1.4914	2.1132	1.8144	1.9436	2.1258	**2.2246**	2.2053
	× 3	22.52	30.04	27.95	30.25	30.86	31.12	**31.51**
		0.5785	0.7725	0.7501	0.7613	0.7701	0.7762	**0.7791**
		1.4307	1.5923	1.5414	1.5562	1.5725	1.6002	**1.6145**
	× 1.2	23.25	-	28.24	-	31.62	31.53	**31.72**
		0.6016	-	0.7768	-	0.7884	0.7796	**0.7962**
		1.4752	-	1.5537	-	1.6365	1.6213	**1.6681**
Urban100	× 2	25.13	**31.75**	28. 72	30.81	31.72	31.61	31.54
		0.6281	0.8146	0.7563	0.8024	0.8168	**0.8195**	0.8191
		1.5241	2.0172	1.8445	1.9195	2.0232	2.0010	**2.1326**
	× 3	23.044	30.63	27.63	28.37	29.74	**30.94**	30.76
		0.5882	0.7954	0.6912	0.7443	0.7811	0.7963	**0.8022**
		1.4307	1.6026	1.5435	1.5678	1.5825	1.5975	**1.6243**
	× 1.2	24.13	-	28.52	-	**31.52**	29.86	31.05
		0. 6016	-	0.7168	-	0.8081	0.7781	**0.8124**
		1.4625	-	1.5527	-	1.6754	1.6454	**1.6809**
Manga109	× 2	24.25	31.29	28.16	30.02	31.30	31.31	**31.83**
		0.6074	**0.8096**	0.7352	0.7816	0.8015	0.8074	0.8053
		1.5018	1.9458	1.8103	1.8931	1.9584	1.9546	**2.0345**
	× 3	21.82	29.88	26.87	28.01	29.08	29.93	**30.87**
		0.5616	0.7815	0.6912	0.7232	0.7576	**0.7983**	0.7791
		1.4233	1.6026	1.5357	1.5557	1.5701	1.5982	**1.6145**
	× 1.2	22.05	-	27.24	-	30.77	30.43	**30.92**
		0.5927	-	0.7085	-	0.7987	0.7981	**0.8065**
		1.4452	-	1.5462	-	1.6143	1.6265	**1.6716**

many image features as possible, achieving the desired visual results of the SR images.

2) COMPARISONS ON DN DEGRADED IMAGES

In this section, we conduct experiments on simulated noise images to verify the performance of the proposed SICNN in the presence of the noise. The noise level σ is set to 10 to imitate a real noise image. Table 2 shows the PSNRs, SSIMs and IFCs obtained from these imitated noise images using the different methods.

We observed that these assessments are all lower than those of the results obtained from the BI degraded images of the previous experiment. Among all these methods, the SICNN still has advantages since it not only has the higher PSNR mean values but it also has relatively high SSIMs and IFCs. The results of the bicubic interpolation method are the lowest. Higher PSNRs indicate that the denoised images are closer to the original clean images; higher SSIMs and IFCs demonstrate that the methods are good at restoring the details such as the edges and textures from a noisy image. For the noisy images with a certain known level, meaning that the paired training dataset is available, the networks have a powerful capability of extracting features, which can provide accurate predictive features for the output data. Thus, the assessment values obtained from the convolutional neural networks (SRCNN, RDN, and meta) are higher than those of the reconstruction-based SR methods (CTV-DNLM). The numerical results from the ZSSR network are considerably better than those of the reconstruction-based SR methods but inferior to those of the SICNN because the ZSSR captures features only from the enlarged images.

Fig.9 and Fig.10 show the visual results from these methods when applied to DN degraded images (img_071 and img_001), respectively. Although the Bicubic method is

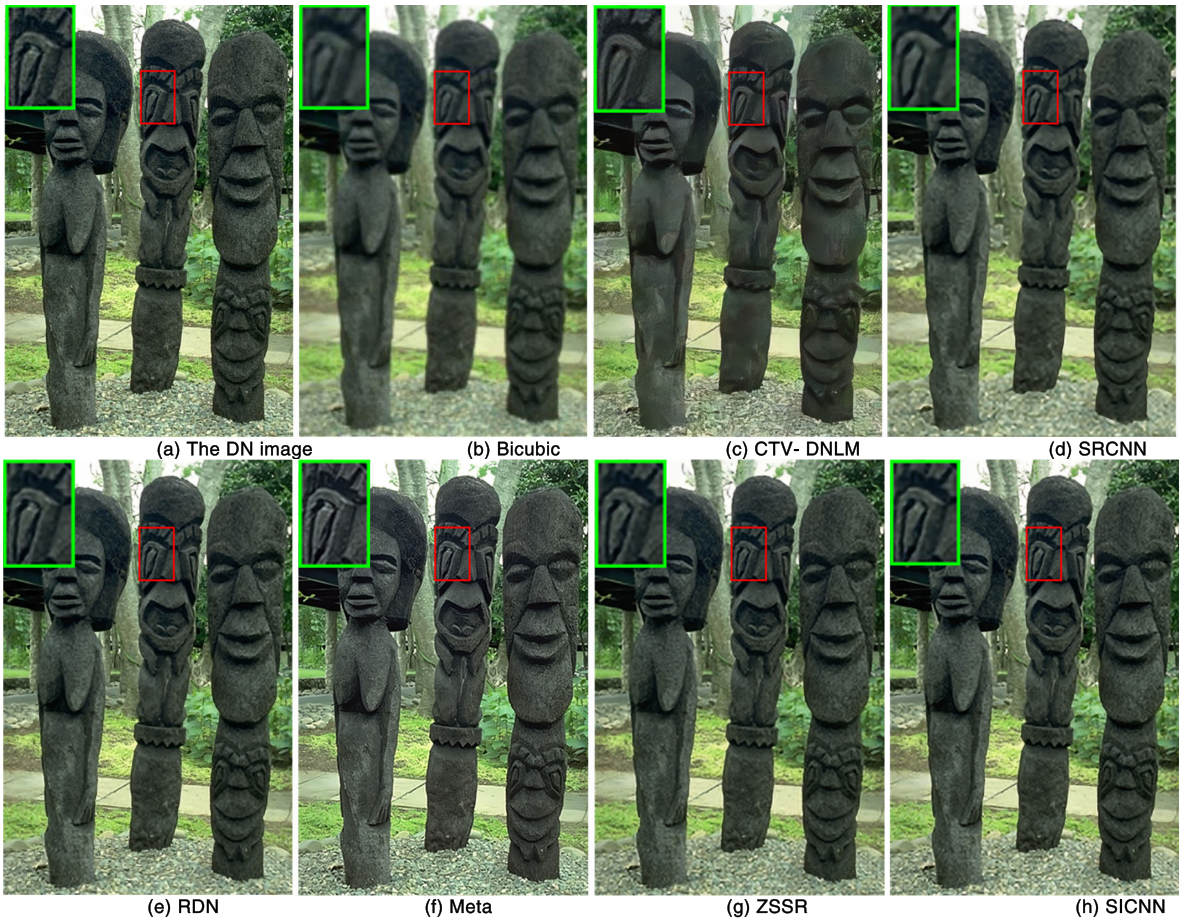


FIGURE 10. Visual comparison of SR results (×2 scale) of DN degraded image *img_001*.

TABLE 3. The average results of NIQE and SSQE from the real-world images with a magnification factor of 3.

Num	Indices	Methods						
		Bicubic	CTV- DNLM	SRCNN	RDN	Meta	ZSSR	SICNN
200	Avg_NIQE	0.9489	0.9432	0.9561	0.9398	0.9174	0.9296	0.9153
200	Avg_SSEQ	7.3251	7.3247	7.3275	7.5639	7.5286	7.4325	7.7502

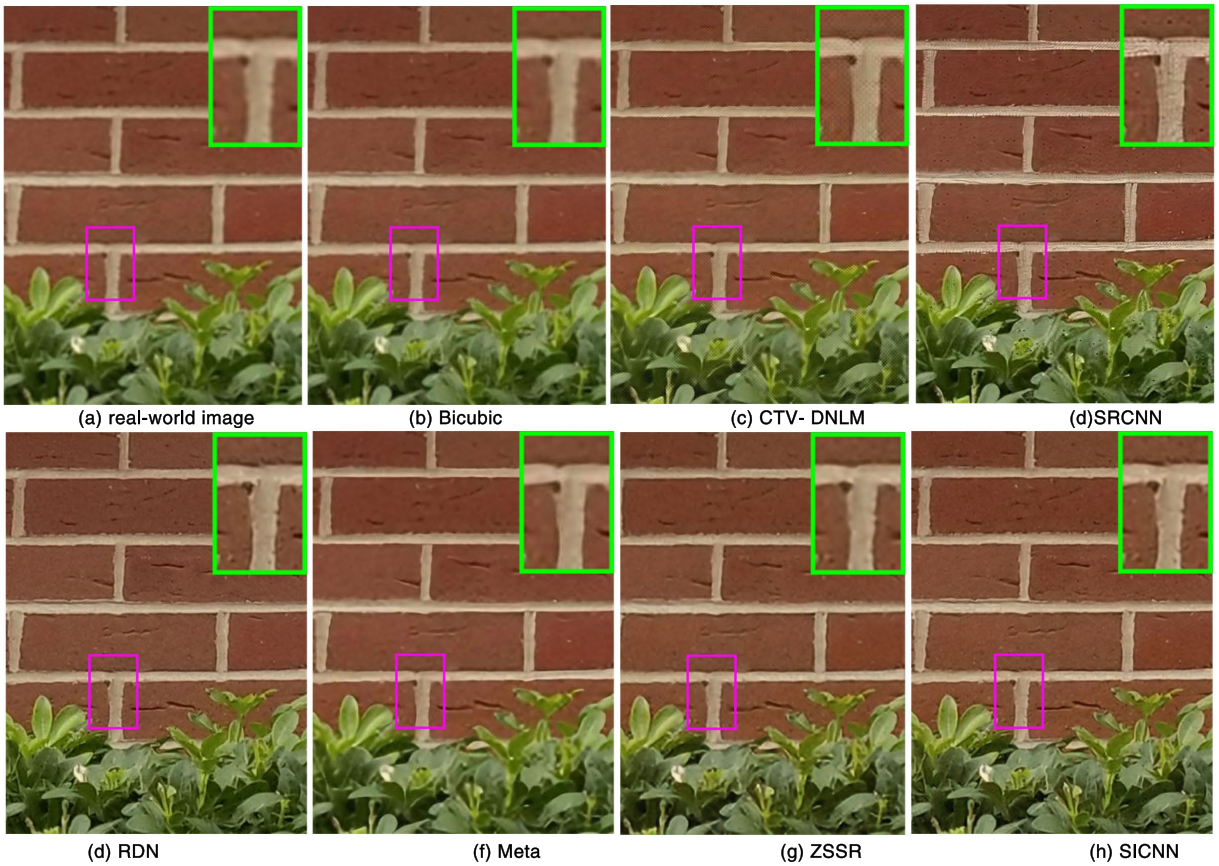
insensitive to noise, the SR images are oversmoothed and lots of details have been lost. The CTV-DMLN produces slight artifacts at the edges despite removing the noise. The SRCNN preserves the rich details of edges but the details are sharpened. The RDN, the meta network and ZSSR network both achieve good quality for image edges and textures in spite of there being a few jaggy artifacts in the smooth regions when they handle slightly smoother images with the amplification factor of an integer multiple of 2(for example, the scale factor is 2). The SR results of the ZSSR network are inferior to those of the meta network but its results are very close to those of the RDN when they handle the images with abundant details with the amplification factor of not an integer multiple of 2 (for example, the scale factor is 3). In contrast, the proposed SICNN achieves a good balance between removing the noise and recovering the detail features of images, achieving satisfactory visual quality regardless of whether the magnification factor is an integral multiple of 2 or not.

3) COMPARISONS ON REAL-WORLD IMAGES

The simulated experiments cannot adequately validate the effectiveness of the proposed method since the degraded images in the simulation are not acquired in a real degradation way. In view of this, we repeat the SR experiments on real-world LR images to illustrate the feasibility and robustness of the proposed SICNN in some real-world scenarios. We randomly collected 260 representative images, which were obtained in the real world under complex degradation conditions such as environmental noise, motion blur and dim light. In this experiment, we focus on the comparisons of the quantitative indicators (NIQE and SSEQ) because there are no corresponding HR images.

(a) House (b) Text

FIGURE 11. Two real-world images and the corresponding ROIs.



(a) real-world image (b) Bicubic (c) CTV- DNLM (d)SRCNN

(d) RDN (f) Meta (g) ZSSR (h) SICNN

FIGURE 12. Visual comparison of SR results (×2 scale) of real-world image *House*.

The quantitative evaluation of the baseline methods and our SICNN, that is, the NIQE and SSEQ indicator values, are shown in Table 3. The NIQE index is used to measure the quality of distorted images and is expressed as a simple distance between the distorted images and the model that is constructed via statistical features collected from a natural scene. The SSEQ index can assess the quality of a distorted image across multiple distortion categories by utilizing the local spatial and spectral entropy features of distorted images. The larger the SSEQ and the smaller the NIQE are, the better the SR reconstruction quality. In Table 3, the SSEQ of the CTV- DNLM method is the lowest while the NIQE of the

FIGURE 13. Visual comparison of SR results (×3 scale) of real-world image *Text*.

SRCNN network is the highest, which are partly because these two schemes cannot estimate the fine details in the textural region. Overall, the metrics for the different methods on the House and Text images suggest that the results of our SICNN are the most satisfactory.

Two ROIs selected from the House and Text images are presented in Fig. 11. Figs.12 and 13 show the SR image qualities obtained from the baseline methods and the proposed SICNN when the images are magnified by 3 times. From the results, we see that the comparison methods bicubic interpolation, CTV-DNLM [11], SRCNN [17], meta [22], RDN [34], and ZSSR [26] have some details lost to some extent or blur some texture details and edges. In contrast, our proposed SICNN can generate clear image details and well restore image textures. The results of our SICNN contain finer textural details and fewer noticeable artifacts along the sharp edges.

V. CONCLUSION

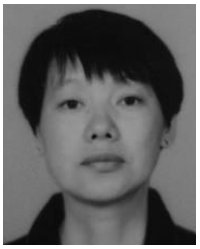
In this paper, we proposed a novel multiscale network, which creates an internal dataset using the test image itself and exploits a double-branch structure to capture and train the image features at different scales, for SR image reconstruction (SICNN). To make full use of the relative information between the local features and the overall features, residual features learning is introduced to the branch structure of large scale-feature mapping to further boost the reconstruction performance. Another difference from the existing SR image reconstructing network is that the SICNN first enlarges the image and then reconstructs it. This means two things: the image can be reconstructed at any arbitrary scale, and the reconstructed feature map can be directly concatenated to form the final image without any more enlargement. The quantitative and qualitative evaluations demonstrate that the proposed SICNN outperforms the existing super-resolution methods from some images with relatively centralized distribution of the gradient values; furthermore, the comprehensive evaluations on benchmark datasets demonstrate that the proposed SICNN achieves performance close to the state-of-the-art super-resolution methods but it is clearly superior to those methods that do not require external datasets.

In future work, we will explore a suitable strategy to share the parameters and thus extend our SICNN model to the SAR image restoration field because it is difficult to obtain appropriate external datasets for SAR images.

REFERENCES

[1] K. Jiang, Z. Wang, P. Yi, G. Wang, K. Gu, and J. Jiang, "ATMFN: Adaptive-threshold-based multi-model fusion network for compressed face hallucination," *IEEE Trans. Multimedia*, vol. 22, no. 10, pp. 2734–2747, Oct. 2020.

[2] X. Gao, L. Zhang, and X. Mou, "Single image super-resolution using dual-branch convolutional neural network," *IEEE Access*, vol. 7, pp. 15767–15778, 2019.
[3] P. Yi, Z. Wang, K. Jiang, Z. Shao, and J. Ma, "Multi-temporal ultra dense memory network for video super-resolution," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 30, no. 8, pp. 2503–2516, Aug. 2020.
[4] X. Li and M. T. Orchard, "New edge-directed interpolation," *IEEE Trans. Image Process.*, vol. 10, no. 10, pp. 1521–1527, Oct. 2001.
[5] X. Zhang and X. Wu, "Image interpolation by adaptive 2-D autoregressive modeling and soft-decision estimation," *IEEE Trans. Image Process.*, vol. 17, no. 6, pp. 887–896, Jun. 2008.
[6] T. Michaeli and M. Irani, "Nonparametric blind super-resolution," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2013, pp. 945–952.
[7] C. Fernandez-Granda and E. J. Candès, "Super-resolution via transform-invariant group-sparse regularization," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2013, pp. 3336–3343.
[8] D. Tao, J. Cheng, X. Lin, and J. Yu, "Local structure preserving discriminative projections for RGB-D sensor-based scene classification," *Inf. Sci.*, vol. 320, pp. 383–394, Nov. 2015.
[9] H. Takeda, S. Farsiu, and P. Milanfar, "Kernel regression for image processing and reconstruction," *IEEE Trans. Image Process.*, vol. 16, no. 2, pp. 349–366, Feb. 2007.
[10] M. Protter, M. Elad, H. Takeda, and P. Milanfar, "Generalizing the nonlocal-means to super-resolution reconstruction," *IEEE Trans. Image Process.*, vol. 18, no. 1, pp. 36–51, Jan. 2009.
[11] X. Li, H. He, R. Wang, and D. Tao, "Single image superresolution via directional group sparsity and directional features," *IEEE Trans. Image Process.*, vol. 24, no. 9, pp. 2874–2888, Sep. 2015.
[12] J. Yu, X. Gao, D. Tao, X. Li, and K. Zhang, "A unified learning framework for single image super-resolution," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 25, no. 4, pp. 780–792, Apr. 2014.
[13] 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.
[14] M. Song, C. Chen, J. Bu, and T. Sha, "Image-based facial sketch-to-photo synthesis via online coupled dictionary learning," *Inf. Sci.*, vol. 193, pp. 233–246, Jun. 2012.
[15] K. In Kim and Y. Kwon, "Single-image super-resolution using sparse regression and natural image prior," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 6, pp. 1127–1133, Jun. 2010.
[16] R. Timofte, V. De, and L. V. Gool, "Anchored neighborhood regression for fast example-based super-resolution," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2013, pp. 1920–1927.
[17] 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.
[18] C. Dong, C. C. Loy, and X. Tang, "Accelerating the super-resolution convolutional neural network," in *Proc. ECCV*, 2016, pp. 391–407.
[19] W. Shi, J. Caballero, F. Huszar, 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.
[20] 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.
[21] 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. Workshops (CVPRW)*, Jul. 2017, pp. 136–144.
[22] X. Hu, H. Mu, X. Zhang, Z. Wang, T. Tan, and J. Sun, "Meta-SR: A magnification-arbitrary network for super-resolution," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 1575–1584.
[23] 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, pp. 1646–1654.
[24] J. Kim, J. K. Lee, and K. M. Lee, "Deeply-recursive convolutional network for image super-resolution," in *Proc. CVPR*, 2016, pp. 1637–1645.
[25] Y. Tai, J. Yang, and X. Liu, "Image super-resolution via deep recursive residual network," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 3147–3155.
[26] A. Shocher, N. Cohen, and M. Irani, "Zero-shot super-resolution using deep internal learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 3118–3126.
[27] M. Bevilacqua, A. Roumy, C. Guillemot, and M.-L.-A. Morel, "Low-complexity single-image super-resolution based on nonnegative neighbor embedding," in *Proc. Procedings Brit. Mach. Vis. Conf.*, 2012, pp. 135.1–135.10.
[28] R. Zeyde, M. Elad, and M. Protter, "On single image scale-up using sparse-representations," in *Proc. 7th Int. Conf. Curves Surf.*, Jun. 2010, pp. 711–730.
[29] P. Arbeláez, M. Maire, C. Fowlkes, and J. Malik, "Contour detection and hierarchical image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 5, pp. 898–916, May 2011.
[30] 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.
[31] M. D. Zeiler, G. W. Taylor, and R. Fergus, "Adaptive deconvolutional networks for mid and high level feature learning," in *Proc. Int. Conf. Comput. Vis.*, Nov. 2011, pp. 2018–2025.
[32] V. Dumoulin and F. Visin, "A guide to convolution arithmetic for deep learning," 2016, *arXiv:1603.07285*. [Online]. Available: http://arxiv.org/abs/1603.07285
[33] 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. ECCV*, 2018, pp. 286–301.
[34] Y. Zhang, Y. Tian, Y. Kong, B. Zhong, and Y. Fu, "Residual dense network for image super-resolution," in *Proc. CVPR*, 2018, pp. 2472–2481.
[35] 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.
[36] J. Li, F. Fang, K. Mei, and G. Zhang, "Multi-scale residual network for image super-resolution," in *Proc. ECCV*, 2018, pp. 517–532.
[37] 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.
[38] L. Liu, B. Liu, H. Huang, and A. C. Bovik, "No-reference image quality assessment based on spatial and spectral entropies," *Signal Process., Image Commun.*, vol. 29, no. 8, pp. 856–863, Sep. 2014.
[39] G. Wang, Z. Wang, K. Gu, L. Li, Z. Xia, and L. Wu, "Blind quality metric of DIBR-synthesized images in the discrete wavelet transform domain," *IEEE Trans. Image Process.*, vol. 29, pp. 1802–1814, 2020.
[40] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2009, pp. 248–255.
[41] K. Ma, Z. Duanmu, Q. Wu, Z. Wang, H. Yong, H. Li, and L. Zhang, "Waterloo exploration database: New challenges for image quality assessment models," *IEEE Trans. Image Process.*, vol. 26, no. 2, pp. 1004–1016, Feb. 2017.



JING LIU received the Ph.D. degree from the Xi'an University of Technology, Xi'an, China, in 2017. She is currently an Associate Professor with the Xi'an University of Technology. Her current research interests include digital watermarking, image processing, and machine learning.

YUXIN XUE received the B.S. degree in computer science and technology from Shangluo University, Shanxi, China, in 2018. She is currently pursuing the master's degree in computer application technology with the Xi'an University of Technology. Her research interests include digital image watermarking and pattern recognition.

SHANSHAN ZHAO received the B.Sc. and M.Sc. degrees in mechanics engineering from Xi'an Technological University, Xi'an, China, in 2001 and 2004, respectively, and the Ph.D. degree in mechanical engineering from Xi'an Jiaotong University, Xi'an, in 2008. She is currently a Visiting Research Fellow with the Department of Engineering, Design, and Mathematics, University of the West of England. Her current research interests include smart manufacturing, computational aerodynamics, industrial safety, and augmented reality.

SHANCANG LI (Member, IEEE) received the B.Sc. and M.Sc. degrees in mechanics engineering and the Ph.D. degree in computer science from Xi'an Jiaotong University, Xi'an, China, in 2001, 2004, and 2008, respectively. He is currently a Senior Lecturer in Cyber Security with the Department of Computer Science and Creative Technologies, University of the West of England, U.K. His current research interests include digital forensics for emerging technologies, network security, cyber attacks, wireless sensor networks, the Internet of Things, and the lightweight cryptography in resource constrained devices.

XIAOYAN ZHANG received the Ph.D. degree from Northwestern Polytechnical University, Xi'an, China, in 2004. She is currently an Associate Professor with the Information Science Technology College, Tan Kah Kee College, Xiamen University. Her current research interests include digital watermarking, image processing, and machine learning.