Spatial Steganalysis Based on Non-Local Block and Multi-Channel Convolutional Networks

XU HAN$^1$ AND TAO ZHANG$^2$

$^1$School of Computer Science and Technology, Soochow University, Suzhou 215006, China
$^2$School of Computer Science and Engineering, Changshu Institute of Technology, Changshu 215500, China

Corresponding author: Tao Zhang (tzhang@cslg.edu.cn)

This work was supported by the National Natural Science Foundation of China under Grant 62072057.

ABSTRACT Image steganalysis aims to detect whether secret information is hidden in images and is a means to solve the communication security. Recently, a series of convolutional neural network-based steganalysis models has been proposed and has achieved remarkable performance. Given that the steganalysis task is different from other computer vision tasks, the series of models is inevitably restricted by manual factors, and some other useful residual information will be missed in the process of feature extraction. To solve this problem, this paper proposes a network model based on non-local operations and multi-channel convolution as part of the basic block of feature extraction for spatial grayscale image steganalysis. The preprocessing layer is improved to compress the image content, further using the introduced non-local operations and multi-channel convolution modules to enhance the residual information in high-frequency regions and extract diverse steganographic features effectively. Extensive ablation study shows that the introduced extraction module can improve the steganographic detection accuracy effectively. Our method is generally better than other methods in terms of detection accuracy, especially in low payload detection.

INDEX TERMS Convolutional neural network, multi-channel convolution, non-local block, steganalysis.

I. INTRODUCTION

Image steganography is used to hide secret information in image areas that are insensitive to the human visual system, such as texture areas and edge areas [1], to deceive the inspector, so as to achieve unknown secret messaging behavior. There are two necessary conditions for the development of image steganography. First, there is a lot of redundant information in the image, and slight changes in the value between adjacent pixel blocks will not affect the final visual cognition (Figure 1). Early steganography will select the lowest pixel as the basis for modification, such as LSBR [2] and OutGauss [3]. Second, with the development of the Internet, increasing number of steganography algorithms and related papers that further promote the use of steganography can be found online.

In contrast, image steganalysis is used to detect whether secret information is hidden in an image [4], [5]. According to the detection results, the images to be detected can be divided into normal images (i.e., cover images) and steganographic images (i.e., stego image), which belong to a binary classification problem. Since steganography modifies certain pixels of the image and results in changing the natural statistics of the original image, for the image content, this modification can be seen as adding a certain noise residual to the image.
content, and steganalysis is subject to suppress the image content while bringing out the noise (payload). Traditional steganalysis task is generally divided into two steps, namely, feature extraction and classification. By extracting the correlation statistics between the image pixels designed by hand in advance and then dividing them by machine learning algorithms, such as subtractive pixel adjacency matrix (SPAM) [6], spatial rich model (SRM) [7], and projection SRM (PSRM) [8]. Although traditional steganalysis has achieved relatively good performance, it also faces several problems. On the one hand, the extracted feature dimension is too high, and some even reach tens of thousands of dimensions, which inevitably increases the computational complexity of the machine. Due to manual design, there will inevitably be redundancy between the extracted features. On the other hand, the extraction of high-dimensional features requires certain expertise to manually design feature models, which inevitably make research difficult.

Several studies have shown that the steganalysis algorithm [9, 10, 11, 12, 13, 14, 15, 16, 17] based on convolutional neural network (CNN) can solve the two aforementioned problems very well. First, CNN realizes the perfect combination of feature extraction and feature classification and completes the end-to-end unified design of steganalysis. Second, feature extraction is completely extracted by the network itself, thereby reducing the difficulty and limitation of manual extraction [18]. However, steganalysis tasks are different from previous computer vision tasks, such as image segmentation, object detection, and image classification, which focus more on extracting the content information of images. On the contrary, steganalysis is more concerned with the embedded residual information. Some researchers have introduced the characteristics of traditional steganalysis in CNN, such as the use of a special filter kernel to initialize the weights of the first layer of the network; the use of a new activation function, such as TLU, to truncate the residual weight, to protect the information of the values in the middle area; and to avoid the loss of residual information, the down-sampling operation is not used in the front part of the network. These methods greatly improve the performance of steganalysis, and a new way of thinking for succeeding research is proposed.

Based on the above analysis, we can find that all network models are designed to suppress image content while bringing out the useful information. However, whether it is the initialization of the preprocessing layer or the use of some novel activation functions, such as TLU and TanH, these designs are all experiences gained from traditional steganalysis. To enhance the embedded information, it will inevitably be restricted by manual design. Recently, Wang et al. [19] introduced the non-local operations into the computer vision (CV) field and proposed the non-local module, which can count non-local long-range dependency information while keeping the input and output size unchanged. Moreover, the result can outperform current competition winners in the video classification task based on the Kinetics and Charades datasets. At the same time, applying this module to object detection and instance segmentation on COCO dataset shows strong performance.

We need to enable the model to pay more attention to the high-frequency and edge regions of the image while extracting features to enhance the residual information of the steganographic embedding. Therefore, this paper proposes an end-to-end neural network for steganalysis based on non-local block. In addition, a large number of experiments compared with the state-of-the-art can prove that our proposed network model not only has a certain improvement in the detection accuracy but also shows a competitive performance on datasets with small payloads.

Motivated by the above observations, our proposed network has the following characteristics:

- In the preprocessing part, we improve the preprocessing layer so that it can autonomously learn the high-frequency embedded regions of the input image for steganalysis;
- In the non-local block part, we introduce the concept of steganalysis based on non-local operations and further enhance the steganographic features by the non-local block to model steganographic noise;
- We introduce a multi-channel nonlinear transformation module and a multi-channel pooling module to enhance the diversity of features.

The remainder of the paper is organized as follows. In Section II, we briefly introduce the related work and development of steganalysis. In Section III, the specific implementation and comparative performance of the network are discussed. In Section IV, the experimental details are discussed, and the experimental results are presented and analyzed. Finally, Section V concludes this paper.

II. RELATED WORKS

Researchers propose more diverse steganalysis schemes [20], [21] to combat increasingly complex steganalysis. Tan and Li [9] applied the deep learning method to the steganalysis task for the first time. The network consisted of a four-layer convolutional autoencoder. The experiments also verified that the randomly initialized CNN usually fails to converge, but it is proved that initializing the weights of the first layer with the KV kernel can improve detection accuracy. The performance of the Tan model is slightly lower than SRM, but it is groundbreaking and opens up a new direction for the succeeding study of steganalysis using CNN. Subsequently, Qian et al. [10] proposed GNCNN. The preprocessing layer is filled with a 5 × 5 high-pass filter that can suppress the interference of image content on steganalysis well and is conducive to strengthening the signal. In the experimental part, the author also pointed out that maxpool only extracts the most important features and will lose some steganographic information. Thus, avgpool is selected for all pooling operations. In terms of performance, GNCNN surpasses SPAM, but is still lower than SRM.
Inspired by the above work, Xu et al. [11] proposed a five-layer network model XuNet, and each convolutional layer contains a BN sublayer and activation function. The network begins from the third convolutional layer, and the subsequent convolutional layers all use $1 \times 1$ convolution kernels, which means that each output is only obtained from the receptive field of the same location of different feature maps. The preprocessing layers of the network are still filled with Gaussian filters, thereby improving the difference between stego image and cover image. Subsequently, Xu [12] further improved XuNet and proposed a network based on decompressed images, which achieved better detection accuracy than traditional methods on the Jpeg domain. Li et al. [13] proposed a ReST-Net network model based on the idea of integrating sub-networks. They believed that the effect of a single branch is not ideal, that is, increasing the network width can extract more diverse steganographic features. Among different network channels, the preprocessing layer of the first channel used 16 Gabor filters [22], the second used 16 linear SRM filters, and the third used 14 nonlinear SRM filters.

Yedroudj et al. [14] proposed the Yedroudj-Net network, which uses 30 SRM filters to fill the first preprocessing layer, to further compress the image content information and introduces the spatial pyramid pooling (SPP) module at the end of the network; hence, it supports the introduction and introduces the spatial pyramid pooling (SPP) module, to further compress the image content infor-

mation and reduces the feature dimension of the connection through the convolution operation, here the residual connection is used; then it will pass through several identical extraction modules (NMC-block $\times 4$ as shown in Figure 2), we name this module non-local and multi-channel convolution block (NMC-block, Figure 3), which include a multi-channel nonlinear transformation module, a non-local block module based on non-local operations and a multi-channel pooling module, and three independent modules are connected in sequence. Inspired by the design of spatial pyramid pooling, we added the SPP structure at the end of the network, after the NMC-block, and before the classification layer, to extract more diverse multi-order steganographic features while reducing feature dimensionality. Pass the classification layer, the classification labels are outputted: cover or stego.

The NMC-block is the focus of this paper. In the previous literature, we found that the main work of a good steganalysis model is to design a feature extraction module. The NMC-block proposed by us consists of three parts. First, it will undergo three parallel convolutions, and the weight are not shared. To increase the feature representation ability, we use different activation functions. Then, the high-frequency region information of the image is enhanced by the non-local block, so that the network pays more attention to the texture region. Finally, the past experience uses the average pooling function or prohibit the use of the pooling layer in the front-end part of the network for down-sampling, but this will cause information loss, so we use avgpool and maxpool in parallel and concatenate the outputs and let the network choose to have representative features. In addition, NMC-block will be used multiple times in sequence, and the parameters of each time are not shared. With the increase in the block, the size of the image features decreases linearly. In each NMC-block, we use a short-cut connection with convolution to avoid network degradation.

III. PROPOSED METHOD
In this section, we present the detailed design of the proposed network framework and key modules. Inspired by non-local operations [19], the network introduces an non-local block to obtain more discriminative steganographic features by designing a multi-channel feature extraction module. In subsequent experiments, for convenience, we call the proposed network as “steganalysis with non-local block and multi-channel convolution” (SNMC-Net).

A. ARCHITECTURE OVERVIEW
The overall framework of the network proposed in this paper is shown in Figure 2. The network mainly consists of three parts: a preprocessing part, multiple consecutive recurrent blocks, and a linear classification module. In particular, our proposed network model is an end-to-end steganalysis network model. By inputting a grayscale image to be detected, the image goes through a preprocessing layer that uses both SRM kernel initialization and random initialization, and guides the extraction of payloads (signal) by simulating the embedded probability map, further compressing the image content (noise) and improving signal-to-noise ratio (SNR), and reduce the feature dimension of the connection through the convolution operation, here the residual connection is used; then it will pass through several identical extraction modules (NMC-block $\times 4$ as shown in Figure 2), we name this module non-local and multi-channel convolution block (NMC-block, Figure 3), which include a multi-channel nonlinear transformation module, a non-local block module based on non-local operations and a multi-channel pooling module, and three independent modules are connected in sequence. Inspired by the design of spatial pyramid pooling, we added the SPP structure at the end of the network, after the NMC-block, and before the classification layer, to extract more diverse multi-order steganographic features while reducing feature dimensionality. Pass the classification layer, the classification labels are outputted: cover or stego.

B. DUAL-CHANNEL PREPROCESS MODULE

1) MOTIVATION
Steganalysis tasks are different from other computer vision tasks. Usually, researchers will add preprocessing modules, such as using HPF [11], SRM kernel [7], and Gabor kernel [22] initialization preprocessing layer, to enhance steganographic features in the network. Other computer vision tasks pay more attention to image content information, whereas steganalysis aims to determine if there is embedded secret information in an image, which is a kind of additive noise relative to the image content. The current steganalysis framework based on deep learning usually has two schemes at the
FIGURE 2. Structure of the "SNMC-Net". For each block $x \rightarrow y$ denotes the number of input and output channels. $\text{Conv}_{a,b,c,d}$ denote the convolution parameters of out_channels, kernel_size, padding, and stride, respectively. Batch normalization is simplified to BN. Fully connected layer is simplified to FC. "concat" means concatenate of output tensor, and "⨁" means residual connect.

preprocessing layer. One is to calculate the residual based on a manually designed initialized filter, the other is to let the network learn the model parameters of the first layer by itself completely.

Based on the above analysis, to suppress the image content information and enhance the noise residual information, we propose a preprocessing layer that uses two filling methods simultaneously. We believe that the problem caused by a specific filter is that the filter is not necessarily optimal, and it may play a role in suppressing part of the dense signal. In Figure 2, the input image will be initialized based on 30 SRM kernels and randomly initialized simultaneously. The specific kernel is used to extract steganographic features, and the random kernel is used to simulate the idea of channel sensing method [15] to generate the embedding probability of the input image.

- Specific initialization: the network takes a $256 \times 256$ grayscale image as input, compresses the image content through a convolutional layer with 30 filters, where the weights of these 30 filters are filled by the SRM kernel, we set the size of all kernels to $5 \times 5$, padding with 0 if the size is not enough. The convolution is followed by a nonlinear activation function TLU and is shown in Equation (1):

$$\text{Trunc}(x) = \begin{cases} 
-T, & x < -T \\
 x, & -T \leq x \leq T \\
 T, & x > T 
\end{cases}$$

where T is set to 3, which can protect the information of the values in the middle area.

- Random initialization: as another branch of the preprocessing layer, random initialization is used to fill the convolution kernel to extract the embedding probability map of the image. The $3 \times 3$ convolution and $1 \times 1$ convolution are used together to reduce the number of parameters when extracting features across channels. An absolute value activation function is interspersed in the middle, which offsets the symbolic symmetry of the features extracted by the preprocessing layer, which is beneficial to enhance the statistical model. The input image is the same as the input for the specific initialization.
• Feature fusion: concatenate the output results of the two branches, the extraction of residual features is guided by the extracted probability embedding map, and use two convolutional layers to reduce dimensionality while compressing image content and improve SNR. The most commonly used ReLU activation function in the field of steganalysis is used here.

2) EXPERIMENTAL VERIFICATION AND ANALYSIS

Through experiments, we demonstrate the advantages of the dual-channel preprocessing layer. We call single-channel as w/o PBC (without preprocessing of basic convolutions, that is, using the SRM kernel to initialize the preprocessing layer alone) and call dual-channel as w/ PBC (with preprocessing of basic convolutions, that is, using both the SRM kernel and random initial kernel to initialize the preprocessing layer). We tested on the BOSSBase1.01 dataset, which is the most widely used in steganalysis. The steganography selects the most commonly used spatial-universal wavelet relative distortion (S-UNIWARD) [23], wavelet obtained weights (WOW) [24], and high-pass, low-pass, low-pass (HILL) [25] among spatial domain adaptive steganography algorithms, and the payloads was selected as 0.4 bpp and 0.2 bpp. In subsequent validation experiments, if not explicitly stated, we will all use the same dataset and steganography.

The experimental results are shown in Table 1. And the expression of the accuracy evaluation metrics is as follows:

\[
\text{Accuracy} = \frac{N_{\text{acc}}}{N_{\text{all}}} \tag{2}
\]

where \(N_{\text{acc}}\) is the number of samples correctly predicted, and \(N_{\text{all}}\) is the total number of samples in test set. For steganography algorithms with different payloads, after using both specific kernel initialization and random initialization, the performance of our network has been greatly improved, especially when using the HILL method at payload of 0.2 bpp, our method has a significant improvement of 3.57%. In addition, when using the S-UNIWARD at payload of 0.2 bpp, the accuracy of our method is also improved by more than 3%. We know that the lower the embedding rate is, the less places there are for modification. By compressing the image content using SRM preprocessing filters and extracting possible modification locations using convolutional simulation, useful information for partitioning can be extracted better.

To further verify the role of dual-channel preprocess module (Figure 4), we visualized the feature maps after using SRM kernel initialization alone and dual-channel initialization. The comparison of Figure 4(b) and Figure 4(c) shows that the latter is significantly darker than the former, and there are two main reasons for this. On one hand, the preprocessing layer initialized with SRM kernel alone usually adds a TLU activation function behind it so that the feature map is truncated to the range of \([-T, T]\), and there will be a lot of bright spots after normalization. On the other hand, after using the dual-channel preprocessing method, we can find that there is less noise in the smooth area of the image.

Furthermore, by comparing the details of the two pictures, the improved method we propose can extract the detailed texture of the face better, this will further compress the image content, thereby highlighting the high-frequency area information of the image.

C. NON-LOCAL MODULE

1) PARADIGM

Adaptive steganography based on the additive model of the distortion function [26], and the syndrome-trellis code (STC) [27] will select the high-frequency texture area of the image as the to-be-embedded area. To make the network pay attention to these regions, Li et al. [28] and Chen et al. [29] used the squeeze-and-excitation method and separable convolution to improve the network performance. We observe that this method only pays attention to the global relationship between convolutional channels and ignores the global weight relationship of the individual feature maps. Therefore, consider the global channel information and local detail information of the feature map simultaneously, we need to design a novel feature extraction module.

The recently proposed non-local block can independently capture the long-term dependencies of each feature map by non-local calculation. The non-local operation is shown in Equation (3):

\[
y_i = \frac{1}{C(x)} \sum_{y_j} f(x_i, x_j) \psi(x_j) \tag{3}
\]
where $i$ is a position index within the spatial feature map whose index response $y_i$ will be computed, and $j$ is an index all location (including $i$ itself), $x$ is the input feature map; and $y$ is the output feature map, which includes the same size as $x$. The binary function $f$ computes the correspondence between each $x_i$ and the other $x_j$, and the function $\psi$ computes the signal of input at position $j$, and the summed result are normalized by $C(x)$.

It is called a non-local operation because when computing the output at position $i$, all possible $j$ related to $i$ are calculated simultaneously. Unlike fully-connected (FC) layers that are often used in classification layers, where non-local operations not only compute output based on all possible locations, but also has a feature that it allows the input feature map to be of random size, and the output size is equal to the input size, but once the number of neurons connected by FC is determined, the input and output size is determined.

The response function $f$ can be defined as Embedded Gaussian, as shown in Equation (4):

$$f(x_i, x_j) = e^{|\theta(x_i)^T\phi(x_j)|}$$  \hspace{1cm} (4)

where $\theta(x_i) = W_\theta x_i$ and $\phi(x_j) = W_\phi x_j$ are two embeddings, and two $1 \times 1$ convolution operations are used. By calculating the relationship of each position to the other positions separately, a matrix multiplication operation, (i.e., $\theta(x_i)^T\phi(x_j)$), is used as it can take full advantage of the fast-computing speed of tensors by hardware devices and directly obtain the response relationship of all positions at one time.

Normalization function $C$: the calculation of the response operation needs to compute the relationship between each position $i$ and any position $j$ when normalizing, that is, $C(x) = \sum_j f(x_i, x_j)$, so standardize $\frac{1}{C(x)}f(x_i, x_j)$ can just be implemented using the activation function $softmax$, and then we proposed the non-local module can be defined as follows:

$$y = W_{softmax}x^T W_{\theta}^T W_{\phi} \psi(x) + x$$  \hspace{1cm} (5)

where $W$ is the parameter of the ascending dimensional convolution, and it can be found here that to accelerate the gradient calculation, the base $e$ is simplified here, and the unary function $\psi$ is also replaced by $1 \times 1$ convolution.

The introduced non-local module is shown in Figure 5. We can observe that the input image will first undergo three $1 \times 1$ convolutions to obtain three feature maps, namely, $\theta$, $\phi$, and $\psi$, and then perform matrix multiplication of $\theta$ and $\phi$, and standardize by $softmax$ to obtain the attention vector. Then, the result of the matrix multiplication is multiplied by the $\psi$ matrix again, and the weighted feature map is outputted. To reduce the number of parameters, all convolution operations in the non-local module are implemented with $1 \times 1$ convolutions. The embedding $\theta$, $\phi$, and the response $\psi$ will reduce the dimension of the feature map, compress the input channel to half, and use $1 \times 1$ convolution to increase the dimension before output. To avoid network degradation, residual connections are used at the output.

The non-local module has the feature that the input and output maintain the same dimension. Here, we opt to put this module in the NMC-block and use it to strengthen the traces of steganography through multiple linear interspersed.

2) EXPERIMENTAL VERIFICATION AND ANALYSIS
To verify the effectiveness of the non-local module, comparative experiments were conducted in the three embedding algorithms, and the experimental results are shown in the Table 2. As a control experiment, we replace the non-local module with a normal $3 \times 3$ convolution module. We can also find that the parameters to be learned for each ordinary $3 \times 3$ convolution are $N \times 3 \times 3$, where $N$ is the number of feature map channels, the number of parameters to be learned by each non-local module is $3 \times N \times 1 \times 1$, and the number of parameters is reduced to 1/3 of the original.

By comparing the results, after the introduction of the non-local module, only under the HILL embedding algorithm at payload of 0.4 bpp, the detection accuracy with the non-local module (our method) is 0.93% lower than without the non-local module. We think this is related to the initialization of network parameters. Among other steganography and pay-loads, the detection results with the non-local module are improved by 0.33%-1.77%, which proves the effectiveness of this module. Non-local block can extract the relationship between feature map channels and different positions within the channel simultaneously, thereby allowing the network to pay more attention to the information in high-frequency regions.

D. MULTI-CHANNEL MODULE
1) MOTIVATION
In the process of feature extraction from the network, to further reduce the feature dimension, performing a
down-sampling operation on the extracted features is necessary. Deng et al. [31] introduced Global Covariance Pooling to further improve the performance of steganalysis, but the pooling calculation is complex, so it is only used once at the end of the network. Common down-sampling operations include max pooling and average pooling. Max pooling can capture the strongest signal of the receptive field but the steganographic signal is extremely weak, and many more important information may be lost. At the same time, average pooling considers all residual information of the receptive field, but reduces the distribution of high-frequency regions. Regardless of the down-sampling method used, a certain amount of steganographic information will be lost.

In image steganography, steganographic traces are embedded into the image texture area with weak residual signals. To reduce the loss of steganographic signals, some studies believe that prohibiting the use of pooling layers in the first half of the network is necessary, and use convolutional layers with kernel size = 3, stride = 2, and padding = 1 instead of pooling layers, which is utilized at the back end of the network. However, the use of convolution increases the number of learnable parameters to learn for the network, whereas the down-sampling function does not. In this paper, we simulate the Xception [30](Figure 6(a)) structure and propose a multi-channel convolution module. To extract more signals of the receptive field, we use two down-sampling methods for the feature extraction of steganalysis without reducing the distribution of high-frequency regions, and the structure is shown in Figure 6 (c).

In addition, as shown in Figure 6 (b), to increase the expressive ability of nonlinear functions and extract more diverse steganographic features, we use three activation functions, namely, ReLU, TanH, and Sigmoid, simultaneously [13]. In the whole network, except for the use of TLU, ABS in the preprocessing layer and NMC-block use three activation functions, and ReLU activation function is used elsewhere.

2) EXPERIMENTAL VERIFICATION AND ANALYSIS

Through the above analysis, we believe that the use of multi-branch in the network can extract diverse feature information. To verify whether our proposed module is effective, we separately conducted two groups of controlled experiments. We simultaneously verified three steganographic embedding methods on the BOSSbase1.01 dataset and compared the network structure without the introduction of the two down-sampling methods (only with avgpool/OAvg, only with maxpool/OMax), and the payload is set to 0.4 bpp. The experimental results are shown in Table 3. The experimental results present that in addition to using the maximum pooling alone on the dataset embedded by the HILL, the accuracy of our proposed method is slightly lower, and the use of the two pooling methods simultaneously has a huge improvement on other embedding methods, especially on the dataset of the WOW steganography. Our method is improved by 1.73% over the average pooling alone. The two pooling methods can consider the weak signals and the distribution of high-frequency regions of the feature map simultaneously and minimize the loss of steganographic signals.

In this subsection, we also experimentally verify the effectiveness of using multiple nonlinear activation functions simultaneously. We believe that the use of each activation function alone will limit the nonlinear expressive power, and the use of multiple activation functions simultaneously can extract richer features. The experimental results are shown in Table 4. By observing the data from the table, the network that

---

**TABLE 2.** Results of non-local operations experiment. Detection accuracy (%) for steganography at 0.2 bpp and 0.4 bpp on BOSSBase1.01 using S-UNIWARD, WOW, and HILL for networks without non-local block (noAtten) and with non-local block (withAtten).

| Algorithms | Payload (bpp) | withAtten | noAtten |
|------------|--------------|-----------|---------|
| S-UNIWARD  | 0.4          | 89.93     | 89.23   |
|            | 0.2          | 78.57     | 77.30   |
| WOW        | 0.4          | 91.73     | 91.40   |
|            | 0.2          | 83.00     | 81.23   |
| HILL       | 0.4          | 84.17     | 85.10   |
|            | 0.2          | 75.10     | 74.73   |

---

**FIGURE 6.** (a). Structure of Xception [30]. (b). Structure of multi-channel nonlinear transformation module. (c). Structure of multi-channel pooling module.
introduces various nonlinear activation functions can obtain higher steganographic detection accuracy under different payloads. Due to the introduction of multi-channel activation functions, the network has 1.73%-3.16% and 0.07%-1.13% improvement on the WOW dataset at 0.4 bpp and 0.2 bpp, respectively. This finding shows that the multi-channel activation function can maximize the use of residual information to extract more discriminative steganographic residual features to improve accuracy.

E. SPP MODULE
To make full use of the feature map context information and enhance the diversity of steganographic information, we use the SPP structure to achieve feature dimension reduction for specific outputs after NMC-block and before full connection. SPP has these advantages:

- For the input images of different sizes, as long as the channel is fixed, it can be outputted with a fixed size;
- The spatial feature information of different scales can be extracted, and the robustness of the model to the distribution of spatial information is improved.

At the end of the network, we add three fully connected layers, and each input image is mapped to a one-dimensional vector of a length of 2. Finally, the softmax activation function is used to implement the probability mapping, thereby representing the probability of the cover category and the stego category.

IV. EXPERIMENTS
A. DATASET
To verify the effectiveness of the proposed network, we conduct multiple sets of fair control experiments, and all experiments are performed on the same dataset. The two standard data sums are presented as follows:

- BOSSbase1.01 [32]: This dataset was captured by seven different cameras and contains 10,000 uncompressed grayscale images of size 512×512;
- BOWS2 [33]: Similar to the BOSSbase1.01 dataset, this dataset also contains 10,000 uncompressed grayscale images of size 512×512. Furthermore, the image distributions of these two datasets are very close.

Due to the limitation of the computing power of hardware equipment, we adjusted the size of the original dataset, MATLAB is used to reduce the original image from 512×512 to 256×256, and all sub-experiments use the same size dataset.

The three most commonly used adaptive steganography algorithms, namely, SUNIWARD, WOW, and HILL, are selected. All embedding operations are implemented through MATLAB, and a random embedding key is used when embedding secret information.

During dataset partitioning, we divided two standard datasets. On the one hand, BOSSbase1.01 standard data set alone was used before dividing the data set. We randomly shuffle the reduced data set and obtain data composed of 10,000 cover images and 10,000 stego images by the steganography embedding algorithm and randomly select 7000 pairs as the training set among 10000 pairs of images and the remaining 3000 pairs as the test and validation sets. The ratio of training, validation, and test set is 7:1.5:1.5, the validation set and the test set are completely distinguished, and there is no overlap. On the other hand, an extended dataset is used (BOWS2 + BOSSbase1.01) to test the generalizability of the network model. The operation is the same as the BOSSbase1.01. After the steganography embedding process, 20000 pairs of data sets containing half of the cover images and half of the stego images will be obtained. We use all BOSSbase1.01 and 4000 pairs of BOWS2 as the training set, and the remaining 6000 pairs of BOWS2 are used as test and validation sets, respectively, and the division is the same as the above data set. All networks will be trained on the training set, model convergence will be observed on the validation set, and model accuracy will be tested on the test set.

B. EXPERIMENTAL DESIGN
The steganalysis network model uses the high embedding rate training model to assist the training of the low embedding rate model, which is naturally in line with the concept of transfer learning. In the training phase, we introduced transfer learning to improve model accuracy and designed three training schemes:

- Scheme #1: Without transfer learning.
- Scheme #2: Pre-train a network model on the high embedding rate training set and then apply the learnable parameters as the initial value of the network parameters to other low embedding rates. In other words, we initialize other model parameters of low payload,
To highlight the advantages of our proposed network, in this article are implemented using the Pytorch framework and can be greatly improved. All experiments involved in this after adding learning rate decay, the accuracy of the network by 0.2 every time 50 epochs are trained. Generally speaking, during the training process, the learning rate is multiplied of the network from oscillating or the loss not decreasing rate is set to 0.01 during initialization. To avoid the accuracy network is trained for a total of 200 epochs, and the learning size during training is set to 32(16 cover/stego). Moreover, computing power and ensure normal operation, the batch for other parameters, is used. To make better use of GPU to 0.0005, and the default parameters of Pytorch are selected parameter momentum is set to 0.9, the weight decay is set during network training, the SGD optimizer, where the method of scheme #3.

is required here, all experiments in this paper use the training the 0.8 payload, so their results are the same. A statement that the downward sequential migration can achieve better are shown in Table5. From the results, it can be concluded such as 0.4 bpp, 0.2 bpp, with parameters pre-trained at 0.8 embedding rate.

- Scheme #3: There is a pre-training model with high payload data set, and which is used to initialize the low payload training parameters. If there is a lower payload training, then migrating downward must be continued. That is, we initialize model parameters of 0.4 bpp with parameters pre-trained at 0.8 payload, and initialize model parameters of 0.2 bpp with parameters pre-trained at 0.4 payload.

To verify which scheme is more efficient, we designed a set of comparative experiments. On the BOSSbase1.01 data set, we use the S-UNIWARD embedding method at payloads of 0.4 bpp, 0.2 bpp, and 0.1 bpp. The experimental results are shown in Table 5. From the results, it can be concluded that the downward sequential migration can achieve better results (i.e. Scheme #3), especially when the payload is small. When the payload is set to 0.1 bpp, the accuracy of training without migration is much lower than using the migration scheme. In addition, when the payload is 0.4, the second and third schemes are both migrated from the model with the 0.8 payload, so their results are the same. A statement is required here, all experiments in this paper use the training method of scheme #3.

C. HYPERPARAMETERS AND PLATFORM

During network training, the SGD optimizer, where the parameter momentum is set to 0.9, the weight decay is set to 0.0005, and the default parameters of Pytorch are selected for other parameters, is used. To make better use of GPU computing power and ensure normal operation, the batch size during training is set to 32(16 cover/stego). Moreover, the loss function selects the cross-entropy loss function. The network is trained for a total of 200 epochs, and the learning rate is set to 0.01 during initialization. To avoid the accuracy of the network from oscillating or the loss not decreasing during the training process, the learning rate is multiplied by 0.2 every time 50 epochs are trained. Generally speaking, after adding learning rate decay, the accuracy of the network can be greatly improved. All experiments involved in this article are implemented using the Pytorch framework and were run on an ubuntu 16.04 with NVIDIA Tesla P100 card.

D. RESULT

To highlight the advantages of our proposed network, in this subsection, we compare with the state-of-the-art networks based on the single BOSSBase1.01 dataset and the scale-up based dataset Bossbase1.01+BOWS2.

1) BASED ON THE BOSSBASE1.01

In Table 6, we report the steganalysis detection accuracy without data augmentation. Through steganographic embedding, the BOSSbase1.01 dataset is randomly divided into 7000 pairs of training set, 1500 pairs of validation set, and 1500 pairs of test set (each pair of images consists of a cover image and its corresponding stego image). As the control group, we selected four network models, namely, Yedroudj-Net, SRNet, Zhu-Net, and GBRAS-Net, and the steganography algorithm selected S-UNIWARD, WOW, and HILL at payloads of 0.4 bpp, 0.2 bpp, and 0.1 bpp. Based on the data in Table 6, our proposed network model significantly outperforms several other representative networks. Except that steganography is S-UNIWARD at payloads of 0.2 bpp and 0.1 bpp, the detection accuracy of our proposed method is slightly lower than SRNet and Zhu-Net, but it still has obvious improvements in other steganography and payloads. We can achieve up to 13.54% improvement over Yedroudj-Net, 1.8% improvement over SRNet, 2.76% improvement over Zhu-Net, and 7.8% improvement over GBRAS-Net. These improvements are benefited from the well-designed network model, including the improved preprocessing layers and NMC-block linked multiple times.

In Figure 7, the loss and detection accuracy curves of datasets based on three different embedding algorithms are shown. We trained the model on the each dataset for 200 epochs, and the training loss, validation loss, training accuracy, and validation accuracy are plotted. When the network is trained to 100 epochs, it basically reaches a state of convergence. To avoid overfitting, we stop training at the 200th epoch. Furthermore, dynamically reducing the learning rate as the training process iterates helps the network to converge further. In the first 100 epochs of the training, the loss and the accuracy of the validation set show an oscillating state, which we think is related to the complexity of the model. Due to the limitation of hardware equipment, we set the batches of network training to 32(16 cover/stego). If the number of batches is too small, the loss function will oscillate and not converge, especially when the network is complex, there are too many parameters to be learned in the network.

The excellence of the proposed model comes from the careful design of the network structure, but this also results in another problem, that is, the complexity of the model and the training time of the model. In Table 7, we show the model size, training time of 200 epochs, and execution time on the test set (1.5K images) for our proposed model and other models. The results in the table show that the training time of the network is not only related to the size of the model but also to the complexity of the network structure. During the iterative training of the network, gradient backpropagation is performed to optimize parameters. The more complex the network structure is, the longer the computation time will be. The number of parameters of our method is not the largest.

| Payload (bpp) | Scheme #1 | Scheme #2 | Scheme #3 |
|--------------|-----------|-----------|-----------|
| 0.4          | 89.13     | 89.93     | 89.93     |
| 0.2          | 78.43     | 77.17     | 78.57     |
| 0.1          | 52.00     | 66.67     | 67.60     |

TABLE 5. Results of transfer learning experiment. Detection accuracy (%) for steganography at 0.4 bpp, 0.2 bpp, and 0.1 bpp on BOSSBase1.01 using S-UNIWARD for three different training scheme.
Our method can extract the residual information of the steganographic embeddings well, regardless of whether it is a high or low payload data, and the experimental results are generally better than all other models. In addition, through the previous ablation experiments, the introduction of multi-channel feature information extraction module and non-local operations enhancement module in the network can improve the network detection performance greatly and modeling steganalysis better, especially in the low payload data.

2) BASED ON THE BOSSBASE1.01 + BOWS2
Data expansion cannot only express the robustness of a model but also improve accuracy and avoid overfitting during training, and augmenting the dataset is a common way to enrich the initial training dataset [34]. In deep learning experiments,

---

**TABLE 6.** Results of the comparison experiment with different networks. Detection accuracy (%) for steganography at 0.4 bpp, 0.2 bpp, and 0.1 bpp on BOSSBase1.01 using S-UNIWARD, WOW, and HILL for five different networks.

| Algorithms   | S-UNIWARD 0.4 | S-UNIWARD 0.2 | S-UNIWARD 0.1 | WOW 0.4 | WOW 0.2 | WOW 0.1 | HILL 0.4 | HILL 0.2 | HILL 0.1 |
|--------------|---------------|---------------|---------------|---------|---------|---------|---------|---------|---------|
| Yedroudji-Net| 78.70         | 65.03         | 57.27         | 84.47   | 72.07   | 61.93   | 76.67   | 67.01   | 55.90   |
| SRNet        | 88.73         | 79.27         | 67.20         | 90.70   | 82.60   | 73.23   | 84.16   | 74.63   | 65.60   |
| Zhu-Net      | 87.37         | 76.40         | 67.90         | 89.40   | 81.50   | 72.27   | 82.53   | 73.67   | 65.87   |
| GBRAS-Net    | 87.10         | 73.60         | 61.73         | 89.80   | 80.30   | 67.23   | 81.90   | 68.30   | 61.20   |
| SNMC-Net     | **89.93**     | 78.57         | 67.60         | **91.73** | **83.00** | **75.03** | **84.17** | **75.10** | **66.60** |

**TABLE 7.** Model parameter comparison. The model parameter size (MB), the training time on 200 epochs, and the execution time on the test set (1.5K images) of the five networks.

| Algorithms   | Model size (MB) | Training time (hours) | Execution time (second) |
|--------------|-----------------|-----------------------|-------------------------|
| Yedroudji-Net| 1.70            | 3.39                  | 4                       |
| SRNet        | 18.22           | 11.64                 | 7                       |
| Zhu-Net      | 10.96           | 4.68                  | 11                      |
| GBRAS-Net    | 0.65            | 6.59                  | 8                       |
| SNMC-Net     | 7.28            | 15.29                 | 15                      |

but due to the complexity of the model structure, the training process consumes the most time.
FIGURE 8. Error probability histogram of the five different networks, namely, SNMC-Net, Yedroudj-Net, SRNet, Zhu-Net, and GBRAS-Net, on two datasets (e.g., BOSSbase1.01 and AugData). (a-e). S-UNIWARD of payloads at 0.4 bpp and 0.2 bpp on BOSSbase1.01 (blue bar) and AugData (orange bar), the horizontal axis represents the various payloads of steganographic algorithms in different datasets, and the vertical axis represents the error probability of the networks (1 means all detections are correct). (f-j). WOW of payloads at 0.4 bpp and 0.2 bpp on BOSSbase1.01 (blue bar) and AugData (orange bar). (k-o). HILL of payloads at 0.4 bpp and 0.2 bpp on BOSSbase1.01 (blue bar) and AugData (orange bar), respectively.

TABLE 8. Results in the expansion experiment. Detection accuracy (%) for steganography at 0.4 bpp and 0.2 bpp on BOSSBase1.01 and BOSSbase+BOWS2(AugData) using S-UNIWARD, WOW, and HILL for five different networks.

| Algorithms | BOSSbase | AugData | BOSSbase | AugData | BOSSbase | AugData | BOSSbase | AugData |
|------------|----------|---------|----------|---------|----------|---------|----------|---------|
|            | 0.4      | 0.2     | 0.4      | 0.2     | 0.4      | 0.2     | 0.4      | 0.2     |
| Yedroudj-Net | 78.70    | 65.03   | 81.93    | 65.72   | 84.47    | 72.07   | 88.78    | 74.00   |
| SRNet       | 88.73    | 79.27   | 91.58    | 81.42   | 90.70    | 82.60   | 92.95    | 85.92   |
| Zhu-Net     | 87.37    | 76.40   | 91.12    | 80.88   | 89.40    | 81.50   | 92.33    | 85.10   |
| GBRAS-Net   | 87.10    | 73.60   | 89.35    | 78.90   | 89.80    | 80.30   | 90.55    | 81.90   |
| SNMC-Net    | 89.93    | 78.57   | 91.90    | 81.40   | 91.73    | 83.00   | 93.10    | 85.73   |

TABLE 8 shows that by expanding the dataset, regardless of the embedding method and payloads, the detection accuracy of our proposed method and the compared methods are improved. For HILL at payload of 0.4 bpp, the detection accuracy of using the expansion dataset is 4.15% higher than that without the expansion dataset, and Yedroudj-Net, SRNet, Zhu-Net, GBRAS-Net are also improved by 0.69%-4.31%, 0.25%-3.80%, 2.93%-4.48%, and 0.75%-6.65%, respectively. Furthermore, in Figure 8, we provide the detection error rates of five steganalysis models under different steganography and payloads. Comparing the detection error rates of each model under the three steganography, the augmented dataset can improve the detection accuracy of the model effectively and further enhance the robustness of the model. In our proposed network, although the improvement in comparing the expansion of the dataset is not the largest, the detection accuracy is still the best in many ways, because of the designed network structure, including the dual-channel in the preprocessing structure, a multi-channel feature extraction module, and an non-local module.
Through the above experiments, for the steganalysis task, we believe that expanding the data set can improve the accuracy of the model and reduce training overfitting effectively and improve the detection performance in the low embedding rate environment.

**V. CONCLUSION**

In this paper, a new CNN-based method for spatial grayscale image steganalysis was proposed on the basis of non-local operations and multi-channel convolution and obtained a better steganographic detection rate than the current CNN-based methods. Our proposed network design is inspired by the following. First, steganographic noise residuals are extracted using a preprocessing module with specific initialization, and embedding probability maps are computed using random initialization. The feature fusion module is used to capture the relationship between the two features. Second, a network module “NMC-block”, which include a multi-channel nonlinear transformation module, an non-local module, and a multi-channel pooling module, is designed for multiple linear connections, and it is the main structure of the network. Third, the introduced multi-channel nonlinear transformation module and the multi-channel pooling module cannot only enhance the diversity of features but also reduce the loss of residual features. Furthermore, the non-local operations can improve the steganographic information in the high-frequency area of the image and enhance the signal-to-noise ratio. These introduced modules can extract steganographic noise further and improve the steganographic detection rate. Considerable research on ablation experiments have been conducted on the BOSSBase1.01 dataset to verify the effectiveness of our proposed module. Meanwhile, in comparison with other representative methods, our proposed network shows significant improvement on detection accuracy. Experiments show that expanding the dataset can improve the detection effect further.

In the future, we will further apply non-local operations to the color image domain and further improve the multi-channel convolution module by studying the relationship among color image differences. Finally, we hope that our research can expand more research thinking for image steganalysis.

**REFERENCES**

[1] A. Cheddad, J. Condell, K. Curran, and P. M. Kevitt, “Digital image steganography: Survey and analysis of current methods,” *Signal Process.*, vol. 90, no. 3, pp. 727–752, Mar. 2010.

[2] I. J. Cox, J. Kilian, T. Leighton, and T. Shamoon, “A secure, imperceptible yet perceptually salient, spread spectrum watermark for multimedia,” in *Proc. Southcon Conf. Rec.*, Jun. 1996, pp. 192–197.

[3] N. Provost, “Defending against statistical steganalysis,” in *Proc. 10th USENIX Secur. Symp.*, 2001, pp. 323–336.

[4] R. Tabares-Soto, R. Ramos-Pollán, G. Isaza, S. Orozco-Arias, M. A. B. Ortiz, H. B. A. Arteaga, A. M. Rubio, and J. A. A. Grisales, “Digital media steganalysis,” in *Digital Media Steganography*. Amsterdam, Netherlands: Elsevier, 2020, pp. 259–293.

[5] T. S. Reinel, R.-P. Raul, and I. Gustavo, “Deep learning applied to steganalysis of digital images: A systematic review,” *IEEE Access*, vol. 7, pp. 68970–68990, 2019.

[6] T. Pevný, P. Bas, and J. Fridrich, “Steganalysis by subtractive pixel adjacency matrix,” *IEEE Trans. Inf. Forensics Security*, vol. 5, no. 2, pp. 215–224, Jun. 2010.

[7] J. Fridrich and J. Kodovský, “Rich models for steganalysis of digital images,” *IEEE Trans. Inf. Forensics Security*, vol. 7, no. 3, pp. 868–882, Jun. 2012.

[8] V. Holub and J. Fridrich, “Random projections of residuals for digital image steganalysis,” *IEEE Trans. Inf. Forensics Security*, vol. 8, no. 12, pp. 1996–2006, Dec. 2013.

[9] S. Tan and B. Li, “Stacked convolutional auto-encoders for steganalysis of digital images,” in *Proc. Signal Inf. Process. Assoc. Annu. Summit Conf. (APSIPA)*, Dec. 2014, pp. 1–4.

[10] Y. Qian, J. Dong, W. Wang, and T. Tan, “Deep learning for steganalysis via convolutional neural networks,” in *Proc. SPIE*, vol. 9409, pp. 171–180, Mar. 2015.

[11] G. Xu, H.-Z. Wu, and Y.-Q. Shi, “Structural design of convolutional neural networks for steganalysis,” *IEEE Signal Process. Lett.*, vol. 23, no. 5, pp. 708–712, May 2016.

[12] G. Xu, “Deep convolutional neural network to detect J-UNIWARD,” in *Proc. 5th ACM Workshop Inf. Hiding Multimedia Secur.*, Jun. 2017, pp. 7–13.

[13] B. Li, W. Wei, A. Ferreira, and S. Tan, “ReST-Net: Diverse activation modules and parallel subnets-based CNN for spatial image steganalysis,” *IEEE Signal Process. Lett.*, vol. 25, no. 5, pp. 650–654, May 2018.

[14] M. Yedroudj, F. Comby, and M. Chaumont, “Yedroudj-Net: An efficient CNN for spatial steganalysis,” in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Apr. 2018, pp. 2092–2096.

[15] M. Boroumand, M. Chen, and J. Fridrich, “Deep residual network for steganalysis of digital images,” *IEEE Trans. Inf. Forensics Security*, vol. 14, no. 5, pp. 1181–1193, May 2019.

[16] R. Zhang, F. Zhu, J. Liu, and G. Liu, “Efficient feature learning and multi-size image steganalysis based on CNN,” 2018, arXiv:1807.11428.

[17] T. S. Reinel, A. A. H. Brayan, B. O. M. Alejandro, M. R. Alejandro, A. G. Daniel, A. G. I. Alejandro, B. J. A. Buenaventura, O. A. Simon, I. Gustavo, and R. P. Raúl, “GBRAS-Net: A convolutional neural network architecture for spatial image steganalysis,” *IEEE Access*, vol. 9, pp. 14340–14350, 2021.

[18] M. Chaumont, “Deep learning in steganography and steganalysis,” in *Digital Media Steganography*. Amsterdam, Netherlands: Elsevier, 2020, pp. 321–349.

[19] X. Wang, R. Girshick, A. Gupta, and K. He, “Non-local neural networks,” in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 7794–7803.

[20] B. Singh, M. Chhajed, A. Sur, and P. Mitra, “Steganalysis using learned denoising kernels,” *Multimedia Tools Appl.*, vol. 80, no. 4, pp. 4903–4917, Feb. 2021.

[21] L. Li, W. Zhang, K. Chen, and N. Yu, “Steganographic security analysis from side channel steganalysis and its complementary attacks,” *IEEE Trans. Multimedia*, vol. 22, no. 10, pp. 2526–2536, Oct. 2020.

[22] X. Song, F. Liu, C. Yang, X. Luo, and Y. Zhang, “Steganalysis of adaptive JPEG steganography using 2D Gabor filters,” in *Proc. 3rd ACM Workshop Inf. Hiding Multimedia Secur.*, Jun. 2015, pp. 15–23.

[23] V. Holub, J. Fridrich, and T. Denemark, “Universal distortion function for steganography in an arbitrary domain,” *EURASIP J. Inf. Secur.*, vol. 2014, no. 1, pp. 1–13, 2014.

[24] V. Holub and J. Fridrich, “Designing steganographic distortion using directional filters,” in *Proc. IEEE Int. Workshop Inf. Forensics Secur. (WIFS)*, Dec. 2012, pp. 234–239.

[25] B. Li, M. Wang, J. Huang, and X. Li, “A new cost function for spatial image steganography,” in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Oct. 2014, pp. 4206–4210.

[26] T. Filler and J. Fridrich, “Gibbs construction in steganography,” *IEEE Trans. Inf. Forensics Security*, vol. 5, no. 4, pp. 705–720, Dec. 2010.

[27] T. Filler, J. Judas, and J. Fridrich, “Minimizing additive distortion in steganography using syndrome-trellis codes,” *IEEE Trans. Inf. Forensics Security*, vol. 6, no. 3, pp. 920–935, Sep. 2011.
[28] J. Li, X. Wang, Y. Song, and S. Wang, “Image steganalysis via squeeze-and-excitation and separable convolutional neural network,” in Proc. 2nd Int. Conf. Artif. Intell., Inf. Process. Cloud Comput. (AHIPCC), Frankfurt, Germany: VDE, Jun. 2021, pp. 1–4.

[29] J. Chen, Z. Fu, X. Sun, and E. Li, “Multi-scale extracting and second-order statistics for lightweight steganalysis,” in Proc. Chin. Conf. Pattern Recognit. Comput. Vis. (PRCV), Cham, Switzerland: Springer, 2021, pp. 548–559.

[30] F. Chollet, “Xception: Deep learning with depthwise separable convolutions,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Jul. 2017, pp. 1251–1258.

[31] X. Deng, B. Chen, W. Luo, and D. Luo, “Fast and effective global covariance pooling network for image steganalysis,” in Proc. ACM Workshop Inf. Hiding Multimedia Secur., Jul. 2019, pp. 230–234.

[32] P. Bas, T. Filler, and T. Pevný, “Break our steganographic system: The ins and outs of organizing BOSS,” in Proc. Int. Workshop Inf. Hiding, Berlin, Germany: Springer, 2011, pp. 59–70.

[33] P. Bas and T. Furon, “BOWS-2 contest (break our watermarking system),” in Organised Within the Activity of the Watermarking Virtual Laboratory (Wavilla) of the European Network of Excellence (ECRYPT), 2008. [Online]. Available: http://bows2.ec-lille.fr/

[34] J. Ni, J. Ye, and Y. I. Yang, “Deep learning hierarchical representations for image steganalysis,” IEEE Trans. Inf. Forensics Security, vol. 12 no. 11, pp. 2545–2557, Nov. 2017.

XU HAN received the B.E. degree from the School of Computer Science and Engineering, Xi’an Technological University, Xi’an, China, in 2020. He is currently pursuing the M.E. degree with the School of Computer Science and Technology, Soochow University, Suzhou, China. His research interests include image processing and information content security.

TAO ZHANG received the M.S. and Ph.D. degrees in signal and information processing from the Zhengzhou University of Information Engineering, in 2000 and 2003, respectively. Since 2019, he has been with the faculty of the Department of Computer Science and Engineering, Changshu Institute of Technology, China, where he is currently a Professor. His research interests include image processing, multimedia information security, and artificial intelligence.

* * *