IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received March 25, 2021, accepted April 12, 2021, date of publication April 19, 2021, date of current version April 27, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3074055

Semi-U-Net: A Lightweight Deep Neural Network for Subject-Sensitive Hashing of HRRS Images

KAIMENG DING[1,2,3], SHOUBAO SU[3], NAN XU[4], AND TINGTING JIANG[5]

[1]School of Network and Communications Engineering, Jinling Institute of Technology, Nanjing 211169, China
[2]State Key Laboratory of Resource and Environment Information System, Institute of Geographic Sciences and Natural Resources Research, Chinese Academy of Science, Beijing 100101, China
[3]Jiangsu Key Laboratory of Data Science and Smart Software, Nanjing 211169, China
[4]School of Intelligent Science and Control Engineering, Jinling Institute of Technology, Nanjing 211169, China
[5]Ericsson (Nanjing) Communications Company Ltd., Nanjing 211100, China

Corresponding author: Shoubao Su (showbo@jit.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 41801303; in part by the Funds for Jiangsu Provincial Sci-Tech Innovation Team of Swarm Computing and Smart Software led by Prof. S. B. Su; in part by the Scientific Research Hatch Fund of Jinling Institute of Technology under Grant jitrcyj-201505, Grant jit-b-201520, Grant jit-fhxm-201604, and Grant D2020005; and in part by the Qing Lan Project.

ABSTRACT As a special case of perceptual hashing algorithm, subject-sensitive hashing can realize ''subject-biased'' integrity authentication of high resolution remote sensing (HRRS) images, which overcomes the deficiencies of existing integrity authentication technologies. However, the existing deep neural network for subject-sensitive hashing have disadvantages such as high model complexity and low computational efficiency. In this paper, we propose an efficient and lightweight deep neural network named Semi-U-net to achieve efficient subject-sensitive hashing. The proposed Semi-U-net realizes the lightweight of the network from three aspects: First, considering the general process of perceptual hashing, it adopts a semi-u-shaped structure, which simplify the model structure and prevent the model from extracting too much redundant information to enhance the robustness of the algorithm; Second, the number of model parameters and the computational cost are significantly reduced by using deep separable convolution in the entire asymmetric network; Third, the number of model parameters is further compressed by using the dropout layer several times. The experimental results show that the size of our Semi-U-Net model is only 5.38M, which is only 1/27 of MUM-net and 1/15 of MultiResUnet. The speed of the Semi-U-Net based subject-sensitive hashing algorithm is 88.6 FPS, which is 2.89 times faster than MultiResUnet based algorithm and 2.1 times faster than MUM-net Based Algorithm. FLOPs of Semi-U-net is only 1/28 of MUM-net and 1/16 of MultiResUnet.

INDEX TERMS Subject-sensitive hashing, lightweight deep neural network, integrity authentication, HRRS image, U-net.

I. INTRODUCTION

The extraction and analysis of earth surface features through high resolution remote sensing (HRRS) images has received extensive research, such as the buildings extraction [1]–[3], vegetation detection [4]–[6], urban expansion analysis [7]–[9] and detection of land cover changes [10]. However, there is a key issue that cannot be ignored: ensuring the security of HRRS image is the basic prerequisite for using HRRS images. If the security of the HRRS image used by the user cannot be guaranteed, the information extracted from the image will be questioned. Among the various security issues of HRRS images, integrity authentication is one of the most sensitive issues.

For the application of HRRS image, if the user uses the tampered images, the analysis result will not be accurate enough, and it is very likely that the wrong analysis result will be obtained. Figure 1 shows comparative examples of HRRS images before and after tampering. Even if the original image is compared, it is not easy to find whether the image has been tampered with. In Figure 1(b), each image from left to right has been tampered with: a building has been added, a building has been deleted, subtle cropping, and random smearing.

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

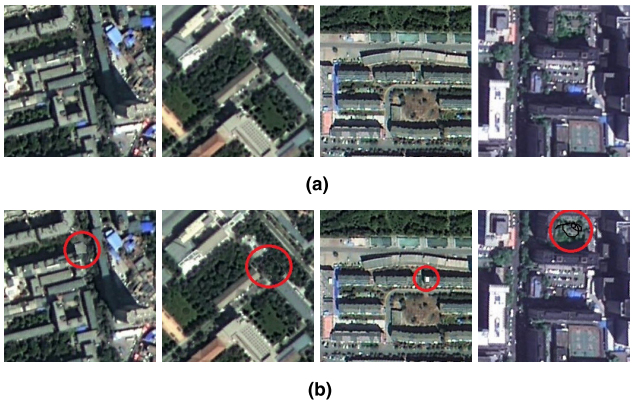


FIGURE 1. A comparative example of HRRS images before and after tampering: (a) Original HRRS images, (b) corresponding tampered images.

Integrity authentication technology can solve the above-mentioned problems, that is, before using HRRS images, integrity authentication technology is used to verify whether the data has been tampered (intentionally or unintentionally) to obtain credible HRRS image data. In other words, integrity authentication technology can ensure the integrity and authenticity of HRRS image, and provide security for the effective use of HRRS image. Traditional image integrity authentication technologies include [11]: cryptography method and watermarking methods. Cryptography methods are mainly based on hash functions or digital signatures to generate image authentication information, and realize image authentication through verification of the authentication information. Digital watermarking generally embeds authentication information into image, and when the content of the image is suspected, the embedded information is extracted and detected whether there is a change.

However, the above kinds of authentication methods have certain shortcomings in the authentication of HRRS image. Cryptography methods are too sensitive to changes in the binary level of the data: As long as the data changes by one bit, it is regarded as data tampering. This sensitivity is detrimental to the integrity authentication of HRRS image. For example, after the lossless data compression of HRRS image, the effective information has not changed, and the availability of the data has not been affected, but cryptography technology believes that the data has been tampered with. Digital watermarking technology will modify the original data more or less, but this modification is not allowed in many occasions. Moreover, digital water-marking technology mainly uses the nature of the watermark itself, and cannot detect whether the effective content of the data has been tampered with. Perceptual hashing can overcome these problems to a certain extent.

Perceptual hash [12], [13], also known as perceptual hashing, is a type of algorithm that can map media data such as images and videos with the same perceptual content to a same string digest, and satisfies security and robustness. Perceptual hash can be further divided into image perceptual hash, video perceptual hash, audio perceptual hash, etc. Image perceptual hash is also called image hashing or robust image hash in some papers. The research of image perceptual hash has been deeply and extensively studied [14]–[19]. Unlike ordinary images that focus on visual effects, HRRS images are more used for information extraction and analysis of features on the earth's surface, often have more stringent requirements for integrity authentication. The design of the perceptual hash algorithms for HRRS images should be based on its data characteristics while taking into account the application environment. At present, the research of perceptual hash of remote sensing images mainly includes the perceptual hash algorithm for HRRS images [20]–[22] and the perceptual hash algorithm of multispectral remote sensing images [23].

In recent years, deep learning is applied to perceptual hash [21], [24]–[29], which has solved many problems in traditional perceptual hash. Subject-sensitive perceptual hash [29], also known as subject-sensitive hashing, is proposed in this background to realize subject-biased integrity authentication of HRRS image. Subject-sensitive hashing consider that different users pay different attention to different image information, However, deep neural network models often have the defects that the model parameters are too large and the calculation cost is too high, which not only leads to a slowdown in the use of the model, but also greatly increases the risk of overfitting. Take the model MUM-net in [29] as an example, the parameter amount of the model is as high as 12 million, and the model after training occupies about 150M of storage space. A model of this scale is not only difficult to apply to mobile devices, but also runs slowly on general servers.

In order to overcome the shortcomings of existing methods, such as excessive storage space and high computational complexity, we propose an effective lightweight deep neural network model Semi-U-net for subject-sensitive hashing of HRRS images. Since the structure of the proposed deep neural network model resembles half of the letter U, it is named Semi-U-Net. The design principle of the model follows the characteristics of subject-sensitive hashing that the redundancy of the extracted features should be as low as possible, and also draws on the idea of deep separable convolution of MobileNets [30].

Our contributions can be summarized as follows:

1. Combining the characteristics of subject-sensitive hash to change the structure of the neural network and reduce the redundancy of the network, which allows the model to avoid extracting too much redundant information to enhance the robustness of the algorithm.

2. Compared with the existing algorithm, our model is effectively compressed to only 5.8M without reducing the performance, and has the potential for deployment on mobile platforms.

3. The computational efficiency of Semi-U-net-based subject-sensitive hashing algorithm is 1.3 to 2.8 times that

of existing algorithms. When using an RTX 2080ti GPU for calculation, it only takes less than 12ms to generate the hash sequence of the HRRS image.

The composition of this paper is as follows. The current related works are described in Section 2. Section 3 discuss the details of our proposed Semi-U-Net and subject-sensitive hashing algorithm. The details of the experiments and discussion are presented in Sections 4. The conclusion is drawn in Section 5.

II. RELATED WORK

Perceptual hash [12], also known as perceptual hashing, can be considered as a subset of generalized hashing: the mapping between perceptual features of images and hash sequences. As for subject-sensitive hashing, we consider it to be a subset of perceptual hash. Perceptual hash originated from digital watermarking technology, in which it is used as embedded watermarking information, and later became an independent technology. Perceptual hash has been widely used in image retrieval [25], [26], [31], [32], image copy detection [33], [34], and image integrity authentication [19]–[23], [35].

Although perceptual hash draws on the design concept of cryptographic hash, it is significantly different from cryptographic hash: Perceptual hash generates a hash sequence based on the perceptual content of the image, while the cryptographic hash (such as MD5 and SHA1) generates a hash sequence based on the binary representation of the image data. Therefore, perceptual hash is more suitable for integrity authentication of HRRS image. For example, after the HRRS image data has undergone format conversion or lossless data compression, the represented content of the image has not changed, but the binary level representation of the image has undergone great changes. In this case, perceptual hash believes that the data has not changed, while the cryptographic hash believes that the data has been tampered with. As there are many similarities between remote sensing images and ordinary images in terms of format and storage, perceptual hash of remote sensing images can refer to the perceptual hash of ordinary images.

The most basic and core problem of perceptual hash is how to effectively express multimedia information such as images, that is, how to extract the perceptual features of images. However, traditional image feature extraction methods are essentially artificially designed features, resulting in certain performance insufficiencies in perceptual hash algorithms: it is not easy to distinguish false features caused by light, fog, etc., which makes the algorithm's tampering sensitivity need to be improved; it is impossible to mine the essential features of remote sensing images in applications, making the perception hash algorithm unable to deal with complex environment; if too much emphasis is placed on the algorithm's robustness, it will be difficult for the perceptual hash algorithm to detect small objects in the image, making the sensitivity to tampering needs to be improved. Deep learning can solve the above problems. In fact, deep learning has been successfully applied not only in image processing, but also in other fields such as fault-tolerant tracking control [36], speech recognition [37], autonomous driving [38], fuzzy fixed-time control problem [39], [40]. This provides us with reference methods for applying deep learning in authentication of HRRS images.

For users of HRRS image, they often pay attention to a certain type of specific information in the remote sensing image. Therefore, the integrity authentication technology of HRRS image should pay more attention to the information that users care about. Subject-sensitive perceptual hash can satisfy this subject-biased integrity authentication.

Subject-sensitive perceptual hash, which was first proposed in [29], can be seen as a special case of perceptual hash and can satisfy subject-biased integrity authentication requirements of HRRS image. Refer to the naming method of perceptual hash, subject-sensitive perceptual hash can also be called subject-sensitive hashing. Subject-sensitive hashing (that is, subject-sensitive perceptual hash) is a one-way mapping that takes into account the types of objects the user is concerned about, and can map the image into a digital summary based on the perceptual content of the image. The core of subject-sensitive hashing is to extract image features that satisfy subject-biased authentication.

However, before the rise of deep learning, the implementation of subject-sensitive hashing was very difficult. This was mainly because most traditional artificially designed features did not have the ability to learn from training samples and could not perform subject-biased integrity authentication. With the powerful feature extraction capabilities, deep learning has been widely studied in the fields of classification, segmentation, and detection, and it also provides a good way to implement subject-sensitive hashing. For subject-sensitive hashing, the function of the deep neural network model is to extract perceptual features for integrity authentication, not for visual effects. On the one hand, the extracted features are required to represent the effective information in the original image as much as possible (especially the information related to applications of remote sensing image). On the other hand, it is required that the extracted perceptual features can be easily compressed and coded to form a perceptual hash sequence.

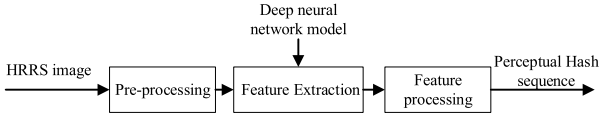


FIGURE 2. Overview of subject-sensitive hashing based on deep neural network.

The overall process of subject-sensitive hashing based on deep neural network is shown in Figure 2, which mainly includes three parts: image preprocessing, subject-sensitive feature extraction, and feature processing. Among them,

the image preprocessing part is to properly process the image according to the algorithm needs to make the processing results more convenient for feature extraction. The preprocessing process mainly performs grid division and resampling of remote sensing images to make the size of the image meet the input conditions of the deep neural network. In the feature extraction part, suitable methods are used to extract the subject-sensitive features of the images according to different application requirements and environments. The remarkable point of subject-sensitive hashing is that the feature extraction method mainly uses deep learning methods, because it is difficult to achieve subject-sensitive hashing based on traditional image feature extraction methods. In [29], MUM-Net undertakes the function of subject-sensitive feature extraction. The feature processing part quantizes and compresses the extracted features so that the final perceptual hash sequence meets the requirements of security and abstraction.

In fact, not only MUM-Net, but other deep neural networks such as U-net [41] and MultiResUNet [42] can also implement subject-sensitive hashing. The above models were compared in [29], and MUM-Net achieves a good balance between robustness and tamper sensitivity, and achieves a better subject-sensitive perceptual hash.

However, the existing deep neural networks that implement subject-sensitive hashing are too complex, the high computing power requirements and energy requirements have become a bottleneck, which makes it difficult to deploy deep school models on mobile terminals (such as drones). Moreover, the computational complexity of calculating the hash sequence of remote sensing images based on MUM-net or other deep neural networks is relatively high, which cannot meet the needs of real-time integrity authentication of remote sensing images. To overcome the above problems, the design of a more efficient neural network is the best way for subject-sensitive perceptual hash to get applications.

In fact, the problem of excessively large deep neural network models is wide-spread. Some scholars have studied lightweight neural networks used in different fields and have achieved good results. In [43], a lightweight neural network is pro-posed for weed mapping tasks, which is 2 times faster than its counterpart. To achieve efficient and real-time small license plate detection on mobile devices, a lightweight model called MobileNet-SSD was proposed in [44]. The lightweight model MobileNet-SSD not only has relatively few parameters, but also improves accuracy. To solve the problem that CNN is overcomplicated in remote sensing image scene classification, a lightweight network based on MobileNet V2 was proposed in [45], which remained relatively high precision. In [46], a lightweight deep neural network model called S2FEF-CNN is proposed for hyperspectral image classification, which can achieve a comparable classification accuracy with significantly reduced parameters. To alleviate the problem of deep 3D-CNN with a huge number of parameters and too expensive calculation cost, a lightweight 3D-CNN framework was proposed in [47] for PolSAR image classification. The proposed framework in [47] introduced pseudo-3D and 3D-depthwise separable convolutions to reduce the redundancy of 3D convolutions. In [48], an efficient light-weight deep neural network is proposed based on dual-path architecture, which also address the issue that most networks for image research involve too many parameters and computational overheads.

In general, existing deep neural network models that implement subject-sensitive hashing, such as U-net and MUM-Net, have certain shortcomings, including:

1. The models are too complex and the trained model is too large.

2. The feature image extracted through U-net and MUM-Net has to be down-sampled in the further processing process to meet the requirements of abstraction.

Different from conventional deep learning tasks such as image segmentation, the deep neural network for subject-sensitive hashing aims to extract subject-sensitive features, which must reflect the content changes of HRRS images. It should be pointed out that the subject-sensitive features extracted by the deep neural network are ultimately used to generate a short sequence, so there is no limit to the size of the output image for subject-sensitive hashing. What's more, the feature image has to be down-sampled in the further processing process to meet the requirements of abstraction. If the size of the feature image we extract does not need to be down-sampled, not only can the computational complexity be reduced, but more importantly, the obtained features should be more robust. After all, the larger the image, the more redundant information. Too much redundant information will greatly affect the robustness of subject-sensitive hashing. However, the input and output images of the existing deep learning models such as MUM-Net and U-net used for subject-sensitive hashing are equal or have little difference in size, which obviously does not take into account the characteristics of subject-sensitive hashing.

Based on the above considerations, combining with the requirement of subject-sensitive hashing that the extracted features should have as little redundant information as possible, and drawing on the idea of depthwise separable convolution of MobileNets [30], we propose a lightweight deep neural network for the realization of subject-sensitive hashing. As the structure of the proposed lightweight deep neural network model resembles half of the letter U, we named it Semi-U-Net.

III. THE PROPOSED NETWORK AND SUBJECT-SENSITIVE HASHING ALGORITHM

In this section, the representation of Semi-U-Net model is present firstly. Then the implementation details of Semi-U-Net based subject-sensitive hashing algorithm are introduced.

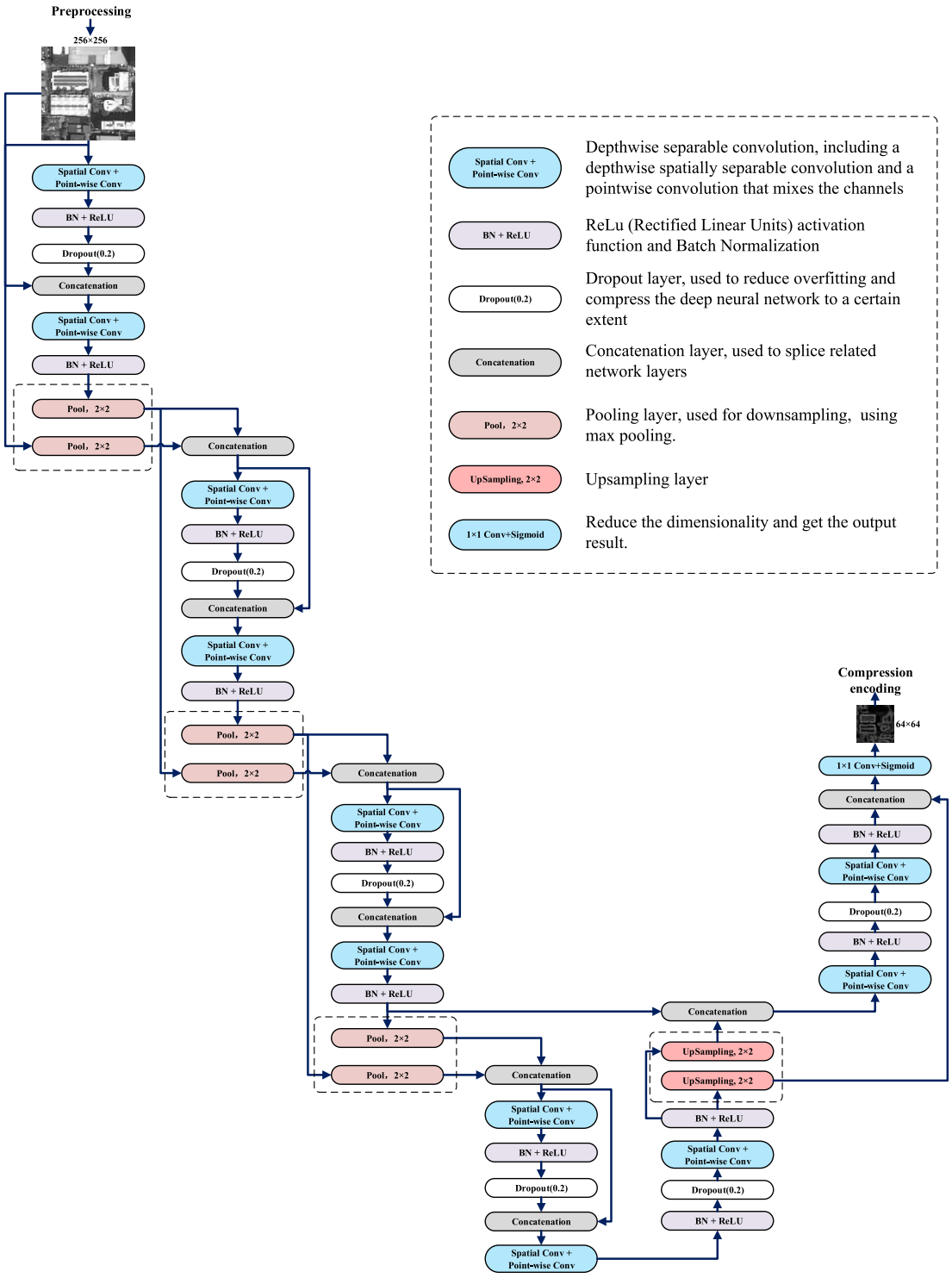


FIGURE 3. Detailed structure of our Semi-U-net architecture.

A. SEMI-U-NET

The architecture of Semi-U-Net is shown in Figure 3. Semi-U-Net is mainly based on the following three aspects to achieve a lightweight model:

1. Replace the convolutional layer by depthwise separable convolution to compress the number of parameters of the network model, which refers to MobileNets;

2. Combining with the general process of perceptual hashing, it adopts a semi-u-shaped structure, which helps to simplify the model structure and prevent the model from extracting too much redundant information and is obviously different from image segmentation network;

3. Since dropout has the characteristics of promoting the sparse distribution of neural network weights [49], we add

dropout in multiple places in the model to compress the neural network with minimal loss of accuracy.

1) ASYMMETRIC SEMI-U-SHAPED NETWORK STRUCTURE

Existing deep neural network models that can implement subject-sensitive hashing (such as U-net, MultiResUnet, MUM-net) extract feature images that are the same or similar in size to the original HRRS image, which inevitably brings redundancy and also increases the complexity of the deep neural network. Therefore, Semi-U-Net adopts asymmetric network structure that is different from U-net, and the structure of the network layer is quite different from U-net in order to achieve a lightweight deep neural network model.

As illustrated in Figure 3, the input for Semi-U-Net is a normalized image with a size of 256 × 256 while the target is a binary mask whose size is one-sixteenth of the input image that is 64 × 64. Similar to traditional image segmentation networks such as U-net and FCN (fully convolutional neural) [50], Semi-U-Net also uses the encoder-decoder structure, which is simple but effective. However, in our Semi-U-Net, the encoder and decoder parts are quite different: the encoder part performs a 3-level pooling operation, while the decoder has only one 1-level upsampling operation.

The encoder phase is composed of a combination of network layers such as depthwise separable convolution, dropout, activation function, batch normalization (BN), pooling (downsampling), and concatenation. Among them, depthwise separable convolution is used to replace traditional convolution operations. In addition, unlike U-net's encoder, the encoder here adds multi-scale input. The entire encoder phase contains 9 depth separable convolutions, divided into 4 groups: the first three groups all contain 2 depth separable convolutions, and the last group contains 3 depth separable convolutions. Except for the last group, after the two depth separable convolutions of each group, one pooling (downsampling) operation is connected. Each depth separable convolution is followed by batch normalization (BN) to speed up the training efficiency and make the network easier to converge. And each use of BN is matched with activation function. In Semi-U-Net, we use the rectified linear unit (ReLU) as the activation function. In order to increase the global feature information, we added multi-scale data input in the encoder part, that is, the original image is directly sampled multiple times, and then merged with the pooling results of the corresponding resolution.

The decoder phase is more different from U-net and FCN. It is consists of depth separable convolution, dropout, ReLU, and BN and upsampling that is not in the encoder, but there is no pooling. Unlike the encoder phase which contains multiple sets of depth separable convolution, the decoder phase has only two set of 2 depth separable convolutions. Using the features map concatenation technique, decoder make up for the features lost in the process of merging and pooling. At the end of decoder phrase, there is a sigmoid function to provide pixel-level classes probabilities.

2) DEPTHWISE SEPARABLE CONVOLUTION

Deep neural networks such as U-net and MUM-Net use standard 2d convolution kernels for convolution operations, and 2D convolution kernels filter the feature maps through the convolution kernel, and then combine the results of different convolution kernels to generate new representations, which requires a complete connection be-tween input and output channels and may make the number of model parameters too large. In our Semi-U-Net model, we refer to MobileNet's depthwise separable convolution to divide the filtering and combination operations into two steps to reduce model parameters and calculations.

Depthwise separable convolution, including a depthwise spatially separable convolution and a pointwise convolution that mixes the channels, performs convolution on each channel separately, and then mixes these outputs through point-wise convolution. This is equivalent to separating the learning of spatial features and channel features. If the input is highly correlated in space and the different accesses are relatively independent, this approach can reduce the number of parameters and reduce the amount of calculation. Our Semi-U-Net uses 3 × 3 depthwise separable convolutions, which is 8 to 9 times less computational than the standard convolution, and the accuracy is only slightly worse.

3) DROPOUT

Since 2017, some scholars have proposed several dropout-based model compression methods. For example, Molchanov [51] used variational dropout to sparse fully connected and convolutional layers, which greatly reduces the number of parameters in standard convolutional networks and has little impact on performance. FPD-M-net [52] also uses dropout many times to compress the number of model parameters. When the number of network layers is the same, FPD-M-net has much smaller amount of parameters than U-net.

In Semi-U-Net model, the dropout layer appears more frequently. Dropout in our model can prevent overfitting and compress the parameters of the neural network model. With reference to the model parameters of FPD-M-net [52], and according to our experimental tests, we set the coefficient of dropout to 0.2.

4) LOSS FUNCTION

The process of extracting subject-sensitive edge features is essentially a binary classification process of pixels in the HRRS image, that is, judging whether each pixel is the edge point of the subject-sensitive edge or not. However, in the training samples of HRRS images, the edges of the object often do not occupy many pixels compared to the non-edge pixels, which means that samples in the classification process are not balanced, that is, the positive samples and negative samples are not balanced. To overcome sample imbalance problem, our Semi-U-Net uses α-balanced variant of focal loss (FL) [53] as loss function, just as MUM-net [29].

The α-balanced variant of FL is as follows:

$$\mathrm{FL}(p_t) = -\alpha_t(1-p_t)^{\gamma}\log(p_t) \tag{1}$$

where α is a value between 0 and 1, and $\gamma \geqslant 0$. The values of α and γ are determined according to the ground features in the training samples. To make our algorithm relatively optimal in terms of robustness and tampering sensitivity, we set $\alpha = 0.25$ and $\gamma = 2$, which is the same as [29] and [53].

B. SEMI-U-NET BASED SUBJECT-SENSITIVE HASHING ALGORITHM

1) PROCESS OF SEMI-U-NET BASED SUBJECT-SENSITIVE HASHING

The deep neural network model is the core of subject-sensitive hashing, but only the deep neural network model cannot constitute a complete subject-sensitive hashing algorithm. Our subject-sensitive hashing algorithm based on Semi-U-Net follows the general steps of subject-sensitive hashing, which consists of preprocessing, subject-sensitive feature extraction, compression encoding, and encryption, as shown in Figure 4:

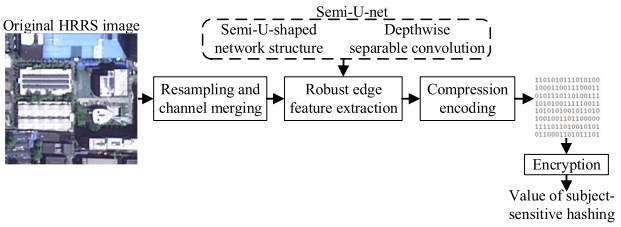


FIGURE 4. Process of Semi-U-Net based subject-sensitive hashing.

1) Preprocessing mainly performs operations such as resampling and channel fusion on HRRS images, so that HRRS images can meet the input requirement of Semi-U-Net. If the HRRS image is too large in practical applications, the method similar to [21], [22], [29] can be used to divide the HRRS image into grids first, and then our algorithm is used. Since our research focuses on lightweight deep neural networks, we do not consider that case.

2) The process of extracting subject-sensitive features is the process of using the trained Semi-U-Net to extract the robust edge features of the preprocessed HRRS image. This process is the core step of subject-sensitive hashing, and the result is a 64×64 grayscale image.

3) In the compression coding process, our algorithm uses conventional PCA-based feature dimensionality reduction and coding methods, and the result obtained is a finite-length binary sequence, which is recorded as *PString*. To improve tampering sensitivity of the algorithm, we adopt the principle that select the high bits of the binary for quantization.

4) The encryption process is to encrypt the binary sequence *PString* through a cryptographic encryption algorithm to ensure the security of the hash sequence itself. In our algorithm, we use the classic AES (Advanced Encryption Standard) algorithm for encryption.

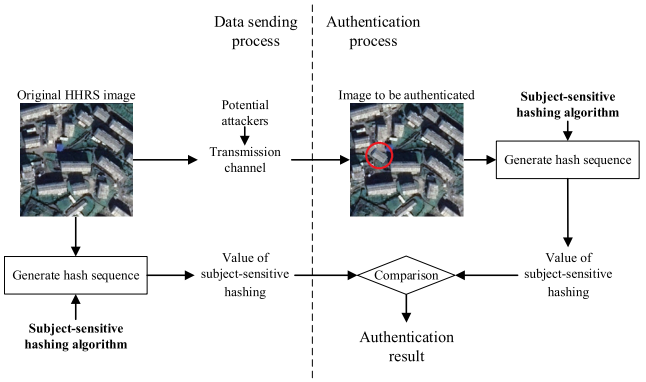


FIGURE 5. Authentication process based on subject-sensitive hashing.

2) AUTHENTICATION PROCESS BASED ON SUBJECT-SENSITIVE HASHING

The integrity authentication process of HRRS images based on our subject-sensitive hashing is shown in Figure 5. The original HRRS image needs to pass through a transmission channel during the transmission process, and the channel may have a potential attacker who causes the HRRS image to be tampered with, as shown in the left half of Figure 5. To realize the integrity authentication of the HRRS image at the data receiving end, it uses the subject-sensitive hashing algorithm to generate the hash value of the HRRS image at data sending end, and sends the generated hash value and the HRRS image to data receiving end.

The integrity authentication process is implemented at the receiving end: the subject-sensitive hashing algorithm is used to generate the hash sequence of the HRRS image to be authenticated, and the generated hash value is compared with the received hash value. In this way, it can be determined whether the content of the received HRRS image has been tampered with. In the process of comparing hash values, we use "normalized hamming distance" [21], [29] to measure the degree of change of the hash sequence. If the normalized hamming distance between the hash values of the original HRRS image and the HRRS image to be authenticated is greater than the threshold T, it means that the content of the HRRS image to be authenticated has been tampered with.

IV. EXPERIMENTS AND DISCUSSION

In this section, we take building information as example of subject to conduct experiments to prove that our method achieves the lightweight of the deep neural network model when the performance is similar to the existing methods. We first briefly describe our experimental environment, including hardware platform and software development

platform. Then, we compare our model with other models to illustrate the advantages of our model. Next, we verify the effectiveness of our model through experiments. Finally, we discuss the performance of our proposed method.

A. SETTING AND DATASETS

1) IMPLEMENTATION DETAILS

Our Semi-U-Net is implemented with Keras 2.3.1 (Tensorflow as backend) and is performed on a machine equipped with a NVIDIA RTX 2080Ti GPU (11 G memory) and an Intel i7-9700K CPU. Due to the limitation of the GPU memory size, the batch size is set to 8, and the epochs is set to 200 during the training of the model. As shown in Figure 6, the training loss curve distribution and the validation loss after each epoch are plotted, where the training data set used will be explained in the later part of this section.

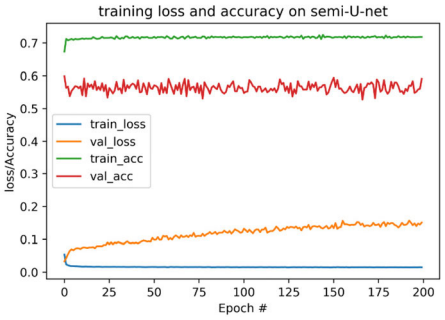


FIGURE 6. Loss and accuracy on semi-U-net during training process.

To maintain the compatibility of the algorithm, the implementation of subject-sensitive hashing algorithm is also implemented with Python. In this way, the Semi-U-Net model after training can be directly applied to the subject-sensitive hashing algorithm.

To evaluate the performance of our model, we compare our model with MUM-Net [29], the original U-net [41], MultiResUnet [42], and M-net [52]. The reason for choosing the above three models as the comparison objects is that our Semi-U-Net draws on the design ideas of U-net to a large extent, and the above four models are closely related to U-net, and they are all in [29] was used to test the subject-sensitive hashing algorithm. Among the above models, MultiResUnet is better in terms of robustness; the original U-net performed better in terms of tampering sensitivity; MUM-Net proved to be a more balanced model with comprehensive performance.

2) DATASETS FOR TRAINING

Like other deep neural network models for subject-sensitive hashing, Semi-U-Net needs to extract subject-sensitive features of HRRS images to generate perceptual hash values. We use the same method in [29] to construct our training data set. The construction of training data set is divided into two steps: First, modify the existing data set to meet the needs of our model; then, draw training samples manually with false edge removed.

The reconstruction of the existing data set is based on the WHU data set [54], which was originally used for building extraction. The original WHU data set contains more than 17,000 training samples with a size of 512 × 512 pixels, collected from different satellite sensors (ZY-3, IKONOS, Worldview series, etc.) with resolutions ranging from 0.3 meters to 2.3 meters. Since our algorithm takes building as the sensitive subject, we did not select all the WDU datasets, but selected 3135 training samples containing buildings to construct our training samples. The selected sample image itself is down-sampled to the size of 256 × 256 pixels, and the corresponding label image is subjected to edge extraction and down-sampling to obtain the label image with a size of 64 × 64 pixels. Moreover, some images that are not selected as training samples are used as test data.

The manual drawing method is to draw robust edge sample images based on GaoFen-2 (GF-2) [55] image. The core work of this is to remove false features in edge images. The size of the original image is resized to 256 × 256 pixels, and the robust edge image is resized to 64 × 64 pixels.

Figure 7 shows examples of training samples. Figure 7(a) is the processed image from the WHU building dataset, whose size was resized to 256 × 256 pixels. Figure 7(b) is the label image corresponding to Figure 7(a). Figure 7(c) is the label image used in our model, with the size of 64 × 64 pixels. Figure 7(d) are images from GF-2, Figure 7(e) are robust edge features generated by artificial processing corresponding to Figure 7 (d), and Figure 7(f) is the label image used in our model, which is the result of the down-sampling of Figure 7(e).

B. COMPARISON OF PARAMETERS AND COMPUTATIONAL RESULTS

1) DATASETS FOR TESTING COMPUTING PERFORMANCE

Since our model focuses on the lightweight of the model and takes into account the enhancement of the calculation speed of the subject-sensitive hash algorithm, we need special datasets to test our Semi-U-Net model and computing performance of the subject-sensitive hash based on this model. In order to avoid the impact of single data set testing, we have constructed 4 test datasets, each of which contains 36, 304, 1,000, and 10,000 HRRS images for testing. The above-mentioned HRRS images for testing are from GaoFen-2 (GF-2) satellite with a spatial resolution of 0.8 m [55], DOTA [56] and WHU building dataset. The size of each image is 256 × 256. We denote these 4 data sets as $Datasets^{36}$, $Datasets^{304}$, $Datasets^{1000}$, $Datasets^{10000}$.

2) COMPARISON OF CALCULATION SPEED

In order to ensure the fairness of the test, we use the same subject-sensitive hashing algorithm process except for the different models for extracting perceptual features. In other

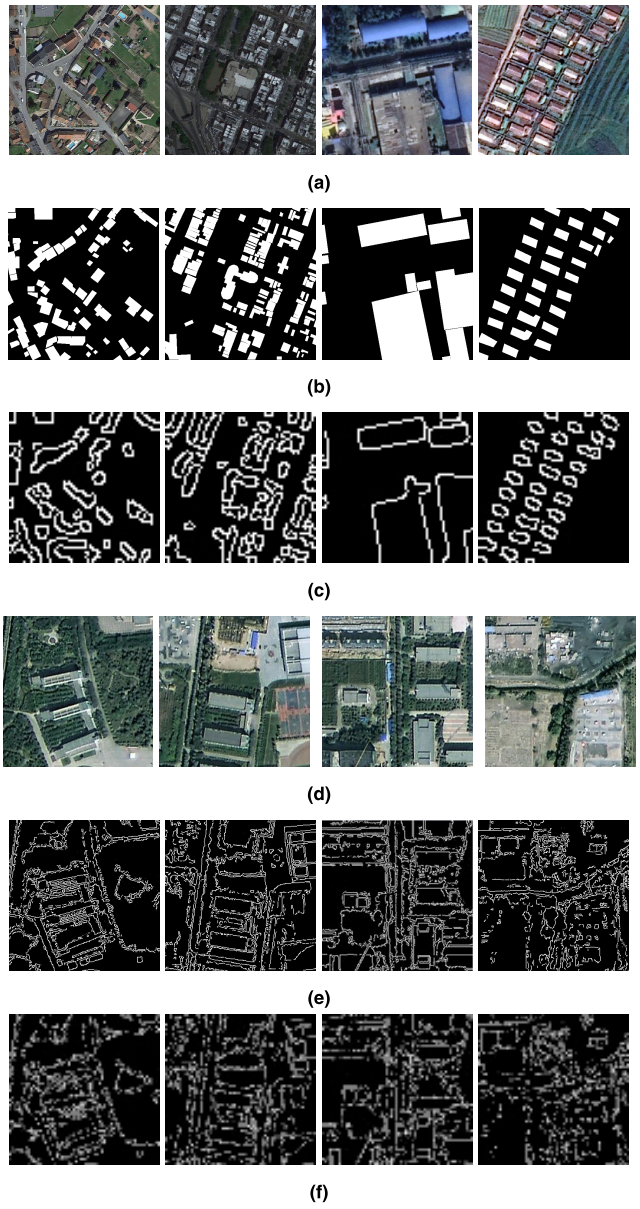


FIGURE 7. Examples of training samples: (a) Processed images from existing datasets, (b) Original labeled images corresponding to (a), (c) Labeled images for semi-U-net corresponding to (a), (d) GF-2 image for artificial training samples, (e) Labeled images generated by artificial processing corresponding to (d), (f) Labeled images used for semi-U-net corresponding to (d).

words, the difference between our comparison algorithms is only the deep neural network model.

For a more intuitive comparison, we evaluate the subject-sensitive hashing algorithms based on different models from three perspectives: total time, average time, and FPS. The comparison results are shown in Table 1.

It can be seen from Table 1 that the Semi-U-Net based algorithm performs best on all data sets. Taking the dataset $Dataset^{10000}$ with 10,000 images as an example, the Semi-U-Net based algorithm only needs 11.29ms to complete the hash calculation of an HRRS image, which is faster than the algorithm based on MUM-Net and MultiResUnet about 2 times and 3 times, and is about 30% and 60% faster than the algorithms based on original U-net and M-net.

From Table 1, it also can be seen that the more images that need to be calculated at one time, the higher the FPS, and the shorter the time to calculate the hash sequence of each image. The reason is that every time to start subject-sensitive hash algorithm, it need to load the model, initialize the GPU, and copy data between the CPU and GPU, even if it only calculate the hash sequence of one image. If the amount of images is small, the longer the average time to calculate the hash sequence, the lower the FPS. Conversely, the more images, the shorter the average time to calculate the hash sequence.

3) COMPARISON OF COMPLEXITY AND STORAGE CONSUMPTION

For the comparison of the complexity of the algorithms, since each algorithm is only different in the deep neural network model while the flow of each algorithm is the same, we focus on comparing each deep neural network model used in the algorithm to illustrate the lightweight and low complexity of Semi-U-net. Here, we compare from three aspects: the number of parameters, the size of the storage space consumed, and Floating Point Operations Per Second (FLOPs). Among them, FLOPs are used to measure the number of floating-point multiplication and addition operations that the convolutional network needs to perform. The results are shown in Table 2.

It can be seen from Table 2 that our Semi-U-Net has significantly reduced the number of parameters, the size of the storage space consumed, and FLOPs compared to the existing model. Compared to the MUM-Net with the best comprehensive performance in [29], the number of parameters and FLOPs of Semi-U-Net is only one 28th of it, and the storage space required is only one 27th of it.

In addition, we can see from Table 2 that the complexity of M-net is also greatly reduced compared to U-net and other models, although it is not as obvious as Semi-U-Net. This shows from the side that the multiple use of the Dropout layer in the design process of our Semi-U-Net plays an important role in the lightweight of the model. Because M-net also uses the Dropout layer many times, and it has more convolutional layers than U-net.

C. PERFORMANCE OF INTEGRITY OF AUTHENTICATION

From the experiments in Section 4.2, it can be seen that subject-sensitive hashing algorithm based on Semi-U-Net has greater advantages over existing algorithms in terms of computing performance, model size, and complexity. However, subject-sensitive hashing algorithms with useful value should not have shortcomings in other performance indicators. In this section, we will compare our algorithm from perspectives of robustness, tampering sensitivity, and security.

TABLE 1. Comparison of computing performance.

	Datasets[36] (36 images)			Datasets[304] (304 images)			Datasets[1000] (1000 images)			Datasets[10000] (10000 images)		
	Total time (s)	Average time (ms)	FPS	Total time (s)	Average time (ms)	FPS	Total time (s)	Average time (ms)	FPS	Total time (s)	Average time (ms)	FPS
MUM-Net Based Algorithm	4.08	113.30	8.8	9.91	32.60	30.7	24.20	24.20	41.3	234.3	23.43	42.7
MultiResUnet Based Algorithm	12.38	343.39	2.9	23.61	77.67	12.9	41.05	41.05	24.4	325.8	32.58	30.7
Original U-net Based Algorithm	3.76	104.40	9.6	6.05	19.90	50.2	16.24	16.24	61.6	146.6	14.67	68.2
M-net Based Algorithm	3.98	110.56	9.0	8.96	29.47	33.9	20.21	20.21	49.5	179.3	17.93	55.8
Semi-U-Net Based Algorithm (Proposed)	2.46	68.33	14.6	5.54	18.22	54.9	14.98	14.98	66.8	112.9	11.29	88.6

TABLE 2. The parameters and storage consumption of the models.

	Input Size	Parameter (M)	Weight Storage (MB)	FLOPs (M)
MUM-Net	256 × 256	12.78	146.64	25.54
MultiResUnet	256 × 256	7.26	83.77	14.55
U-net	256 × 256	9.24	105.86	18.48
M-net	256 × 256	3.54	40.71	7.07
Semi-U-Net	256 × 256	0.45	5.38	0.91

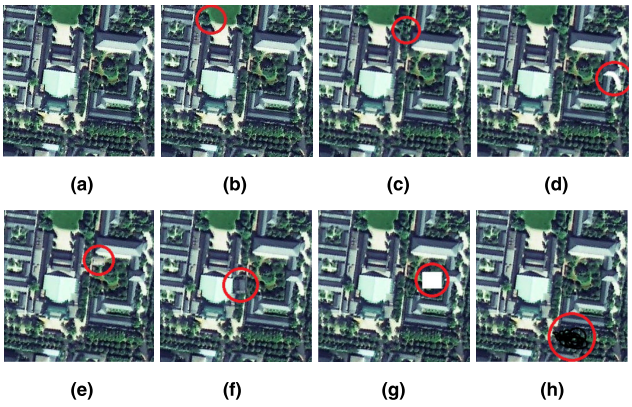


FIGURE 8. Examples of Integrity Authentication for HRRS image: (a) The original HRRS image; (b)-(d) subject-unrelated image content changes; (e)-(h) subject related image content changes.

1) EXAMPLES OF INTEGRITY AUTHENTICATION

To make the comparison of integrity authentication more intuitive, we take a set of examples to compare algorithms based on different models, as shown in Figure 8.

There are two types of tampering examples in Figure 8: subject-related tampering and subject-unrelated tampering. Figure 8(b)-Figure 8(d) are subject-unrelated tampering: the lakeside color of Figure 8(b) has been lightened; Figure 8(c) has been added a tree; the color of the lake in Figure 8(d) has been changed to pure white. Figure 8(e)-Figure 8(h) are subject-related tampering: Figure 8(e) and Figure 8(f) have been added with buildings; Figure 8(g) has been deducted part of area; Figure 8(h) is partially smeared.

We compare the algorithms based on Semi-U-Net with the algorithms based on U-net, M-net, MUM-Net, and MultiResUnet, and set threshold of the normalized hamming distance to 0.03. The results are shown in Table 3.

For subject-related image content changes, the subject-sensitive hash algorithm should maintain a certain tolerance; but for subject-related image content changes, the subject-sensitive hash algorithm must be able to detect, because subject-related image content changes will have a great negative impact on the use value of HRRS images. As for the subject-related image content changes, we believe that the HRRS image has been maliciously tampered with.

It can be seen from Table 3 that the subject-sensitive hashing algorithm based on Semi-U-Net can realize subject-biased integrity authentication like other subject-sensitive hashing algorithms. Specifically, for the subject-unrelated image content changes in Figure 8(b)-Figure 8(d), these five deep learning based algorithms are all robust to a certain degree, while our Semi-U-Net based algorithm is better than U-net based algorithm in robustness, but inferior to MUM-net based algorithm. The image content changes in Figure 8(e)-Figure 8(h) are subject-related, and most of these

TABLE 3. Integrity authentication results of the algorithms based on different models.

Tampering Test	U-net Based Algorithm	M-net Based Algorithm	MultiResUnet Based Algorithm	MUM-net Based Algorithm	Semi-U-Net Based Algorithm
Figure 7(b)	0.0221	0.0013	0.0	0.0013	0.0169
Figure 7(c)	0.0052	0.0026	0.0052	0.0078	0.0036
Figure 7(d)	0.0768	0.0781	0.0755	0.0195	0.0638
Figure 7(e)	0.0521	0.0247	0.0286	0.0599	0.0911
Figure 7(f)	0.0794	0.0572	0.0143	0.0938	0.0625
Figure 7(g)	0.0924	0.0716	0.0898	0.0859	0.0755
Figure 7(h)	0.0755	0.0182	0.0078	0.0442	0.0794

tampering can be detected by each algorithm, except that the MultiResUnet based algorithm failed to detect the tampering in Figure 8(e) and Figure 8(d). Moreover, our Semi-U-Net based algorithm does not lag behind MUM-net based algorithm and U-net based algorithm in terms of tampering sensitivity, and these two algorithms have been proved to have good tampering sensitivity in [29].

In general, the integrity authentication performance of our algorithm does not lag behind the existing subject-sensitive hashing algorithm and has the advantages of light weight and high computing performance.

2) PERFORMANCE OF ROBUSTNESS

The robustness of subject-sensitive hashing requires a large number of examples to prove, which the same as perceptual hashing is. In this section, we use the data set $Datasets^{10000}$ of section 4.2.1 to test the robustness of each comparison algorithm. This data set contains 10,000 images in tiff format, which can meet the needs of robustness testing in quantity.

First, we take data compression as an example to test the robustness of the algorithm. Regarding the process of data compression, we write programs in combination with C++ and OpenCV under Microsoft's visual studio platform to compress the data in TIF format in Datasets. The compression method uses PNG, and the compression level is 8 (compression level is from 0-9, and the larger the compression level, the higher the compression rate). Before data compression, the size of $Datasets^{10000}$ is 1.88G; after data compression, the size of $Datasets^{10000}$ is 1.2G.

The results of the robustness test are shown in Table 4: Under different thresholds, the change of the hash value of each image is less than the threshold (or there is no change). It can be seen that each algorithm has good robustness to PNG compression.

Digital watermarking technology plays an important role in the copyright protection of HRRS images, but different watermark embedding algorithms will have different effects due to factors such as the amount of embedded information, differences in embedding methods, and image size. Moreover, different watermarking algorithms modify HRRS images to different degrees. To illustrate the robustness of each algorithm more comprehensively, we separately test the robustness of each algorithm to "watermarking methods with different amounts of information embedded"': the original 256 × 256 HRRS Images are embedded with 8-bit, 16-bit, and 32-bit watermark information. The results are shown in Table 5 to Table 7.

From the results in Table 5 to Table 7, it can be seen that in the case of relatively little embedded information, subject-sensitive hashing algorithms based on different models have better robustness: When the embedded information is only 8 bits, the robustness of each algorithm is relatively good even if the threshold is set as low as 0.01, and the gap between the algorithms is not obvious. As the embedded information increases, the robustness of the algorithm begins to decline. Overall, the robustness of the algorithm based on MultiResUnet is the best, which is the same as the conclusion of [29]. Our Semi-U-net-based subject-sensitive

TABLE 4. Robustness test comparison of data compression.

	U-net Based Algorithm	M-net Based Algorithm	MultiResUnet Based Algorithm	MUM-net Based Algorithm	Semi-U-Net Based Algorithm
T=0.01	100%	100%	100%	100%	100%
T=0.02	100%	100%	100%	100%	100%
T=0.03	100%	100%	100%	100%	100%
T=0.05	100%	100%	100%	100%	100%
T=0.1	100%	100%	100%	100%	100%

TABLE 5. Robustness test comparison of digital watermark embedding (8-bit information embedding).

	U-net Based Algorithm	M-net Based Algorithm	MultiResUnet Based Algorithm	MUM-net Based Algorithm	Semi-U-Net Based Algorithm
T=0.01	95.0%	97.2%	98.1%	94.5%	93.6%
T=0.02	98.9%	99.2%	99.8%	98.5%	98.5%
T=0.03	99.4%	99.7%	99.9%	99.6%	99.2%
T=0.05	99.9%	100%	100%	99.7%	100%
T=0.1	100%	100%	100%	100%	100%

TABLE 6. Robustness test comparison of digital watermark embedding (16-bit information embedding).

	U-net Based Algorithm	M-net Based Algorithm	MultiResUnet Based Algorithm	MUM-net Based Algorithm	Semi-U-Net Based Algorithm
T=0.01	89.2%	94.6%	96.0%	88.8%	89.5%
T=0.02	96.0%	98.7%	98.6%	96.8%	98.0%
T=0.03	98.6%	99.6%	99.5%	98.1%	99.3%
T=0.05	99.6%	99.8%	99.9%	99.3%	99.9%
T=0.1	100%	100%	100%	100%	100%

TABLE 7. Robustness test comparison of digital watermark embedding (32-bit information embedding).

	U-net Based Algorithm	M-net Based Algorithm	MultiResUnet Based Algorithm	MUM-net Based Algorithm	Semi-U-Net Based Algorithm
T=0.01	79.8%	87.7%	93.3%	80.2%	79.1%
T=0.02	93.0%	96.6%	98.3%	94.1%	95.2%
T=0.03	96.7%	98.8%	99.4%	97.7%	98.5%
T=0.05	98.9%	99.6%	99.9%	99.2%	99.4%
T=0.1	100%	100%	100%	100%	100%

hashing algorithm is not as robust as the MultiResUnet-based algorithm, but it is better than the U-net-based algorithm.

Next, we simulate the subtle changes of the HRRS image by randomly setting some pixels to 0, and then compare the robustness of each algorithm. If too few pixels are changed (for example, only 1-2 pixels are changed), the content of the HRRS image can be considered unchanged, and the robustness of the algorithm cannot be tested; but if too many pixels are changed, it will cause damage to the content of the HRRS image. For each HRRS image in the test data set, we randomly selects 10 pixels to modify the pixel value to 0. The comparison result of the robustness test is shown in Table 8.

TABLE 8. Robustness test under different threshold for pixel-level changes.

	U-net Based Algorithm	M-net Based Algorithm	MultiResUnet Based Algorithm	MUM-net Based Algorithm	Semi-U-Net Based Algorithm
T=0.01	56.4%	71.4%	86.4%	63.5%	56.8%
T=0.02	75.8%	87.2%	96.0%	82.8%	81.2%
T=0.03	84.5%	92.9%	98.2%	90.8%	91.9%
T=0.05	92.4%	95.8%	99.0%	95.8%	96.8%
T=0.1	99.7%	100%	100%	99.7%	100%

From the results in Table 8, we can see that our Semi-U-net-based subject-sensitive hashing algorithm is more robust to subtle tampering than the U-net-based algorithm. Our algorithm is not much different from the M-net-based algorithm and the MUM-net-based algorithm, but it is not as good as the MultiResUnet-based algorithm. The above analysis results are basically the same as those obtained in Table 5 to Table 7.

It should be pointed out that, as each of the subject-sensitive hash algorithm in this paper adopts the principle of ''selecting the high bits of the binary for quantization'' in the quantization process from the principal component to the hash value, the robustness of the subject-sensitive hash algorithm based on U-net is less than the corresponding algorithm in [29].

In general, although our Semi-U-Net-based algorithm is more robust than U-net-based algorithms, its robustness needs further improvement compared to MultiResUnet based algorithm, MUM-net based algorithm and M-net based algorithm.

3) PERFORMANCE OF SENSITIVITY TO TAMPERING

As a special type of perceptual hash algorithm, subject-sensitive hash must detect possible malicious tampering of HRRS images, otherwise integrity authentication will lose its meaning. Different from traditional authentication algorithms, subject-sensitive hashing is more sensitive to subject-related tampering. Therefore, sensitivity to tampering (also known as ''tampering sensitivity'') is very important for measuring the integrity authentication performance of subject-sensitive hashes. Compared with traditional authentication algorithms, subject-sensitive hashing is more sensitive to subject-related tampering.

Evaluating the tampering sensitivity of subject-sensitive hashing algorithms requires a certain number of test cases to be convincing. We select a part of the existing tampering sensitivity test data (mainly from [21] and [29]), and create tampering examples based on GF-2 satellite images, a total of 800 tampering examples, including adding objects, removing objects, modifying objects, painting on images, etc. Figure 9 shows a set of tampering examples, in which Figure 9 (a) is the original HRRS image,

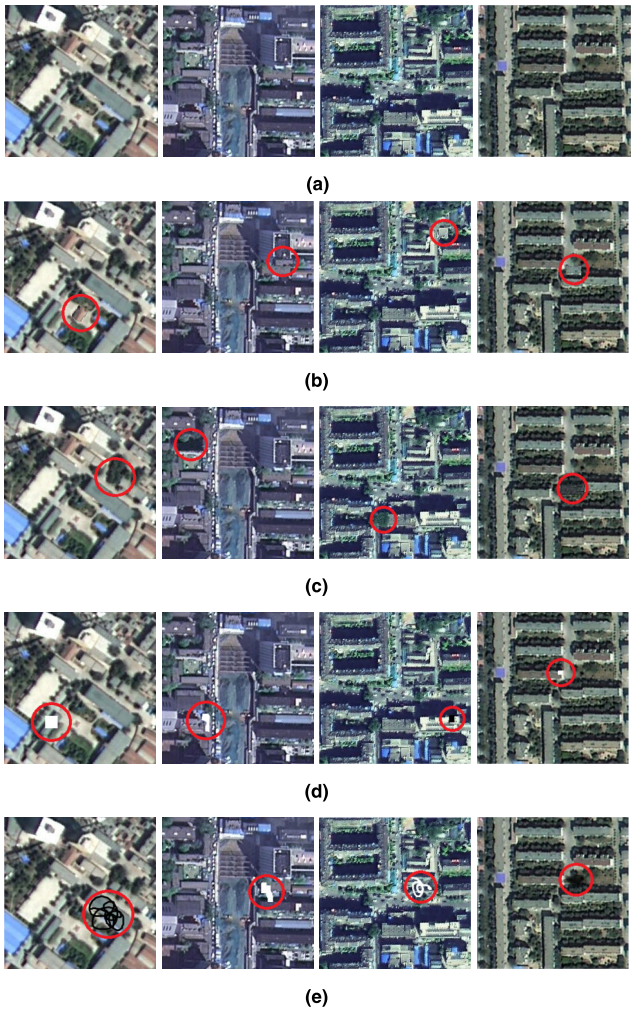


FIGURE 9. Example of subject related tampering: (a) Original HRRS images; (b) adding objects; (c) removing object; (d) modifying objects; (e) smear on image.

Figure 9 (b), Figure 9 (c), Figure 9 (d), Figure 9 (e) are tampering examples such as adding objects, removing objects, modifying objects, painting on images.

For the various types of tampering mentioned above, we use the proportion of tampering detected successfully under different thresholds to describe the tampering sensitivity of the algorithm. The results are shown in Table 9.

It can be seen from Table 9 that the tamper sensitivity of each algorithm based on deep neural networks is related to the setting of the threshold: the smaller the threshold is set, the more conducive to the detection of tampering. Although the original intention of our Semi-U-net-based subject-sensitive hashing algorithm is to implement a lightweight algorithm, the tamper sensitivity of our algorithm is basically similar to that of the subject-sensitive hashing algorithm based on U-net, and it is much better than MultiResUnet based algorithm.

In fact, the tamper sensitivity of each subject-sensitive hash algorithm based on deep neural networks can be changed by the training data set. If there are higher requirements for tampering sensitivity in practical applications, while a lower threshold cannot be set (so as not to affect the robustness of the algorithm), the ability of the model to extract features can be improved by adding high-quality training samples.

TABLE 9. Comparison of the proportion of tampering detected under different thresholds.

	U-net Based Algorithm	M-net Based Algorithm	MultiResUnet Based Algorithm	MUM-net Based Algorithm	Semi-U-Net Based Algorithm
T=0.01	98.8%	95.6%	80.3%	91.1%	96.4%
T=0.02	91.5%	85.8%	71.8%	89.4%	90.3%
T=0.03	87.3%	82.9%	68.9%	81.3%	85.1%
T=0.05	79.1%	78.6%	44.8%	77.5%	80.5%
T=0.1	26.8%	20.2%	14.1%	21.3%	23.5%

D. ANALYSIS OF ALGORITHM SECURITY

Compared with traditional perceptual hash algorithms without using deep learning, the security of Semi-U-Net based subject-sensitive hashing algorithms largely depends on the uninterpretability of deep learning models, which is similar to MUM-Net based algorithm or the perceptual hash algorithm based on U-net based algorithm. In addition, in the coding stage of perception features, our algorithm uses the AES algorithm to ensure the security of the perception hash sequence itself (users without a key cannot perform integrity authentication), which is the same as the existing subject-sensitive hash algorithm. In general, our Semi-U-net-based subject-sensitive hashing algorithm is the same as the existing algorithm in terms of security, and there is no obvious gap or advantage.

E. DISCUSSION

Subject-sensitive hashing takes advantage of deep learning to obtain the ability to extract subject-sensitive features through subject-related samples learning. Therefore, subject-sensitive hashing is essentially the application of deep learning in HRRS image security.

Existing deep neural networks used to implement subject-sensitive hashing, such as U-net and MUM-Net, have disadvantages such as too complex models and large storage space. In this paper, we designed a lightweight Semi-U-net to implement an efficient lightweight subject-sensitive hashing algorithm, which combines the characteristics of perceptual hashing and uses depthwise separable convolutions to replace traditional convolution operations. Summarizing the experimental results of this paper, we can draw the following conclusions:

Firstly, our Semi-U-net has greater advantages over existing models in terms of model complexity and model size: The size of Semi-U-net is only 1/27 of MUM-net, 1/19 of

U-net, 1/15 of MultiResUnet and 1/7.5 of M-net; The FLOPs of Semi-U-net is only 1/28 of MUM-net, 1/20 of U-net, 1/16 of MultiResUnet, and 1/7.7 of M-net. Therefore, our Semi-U-net has achieved the goal of being lightweight.

Secondly, we test and compare the calculation speed of the algorithm on several datasets containing different numbers of images in section 4.3.1. The calculation speed of Semi-U-net based subject-sensitive hashing algorithm is the highest among all data sets, and the larger the amount of data, the shorter the average time it takes to calculate each image. In the fastest case, our Semi-U-net Based Algorithm is 2.89 times faster than MultiResUnet Based Algorithm, 2.1 times faster than MUM-net Based Algorithm, 1.6 times faster than M-net Based Algorithm and 1.3 times faster than U-net Based Algorithm. Therefore, compared with existing algorithms, our Semi-U-net based algorithm has advantages in terms of computational performance.

Third, in terms of robustness, our Semi-U-Net based algorithm is better than U-net based algorithm, but inferior to MultiResUnet based algorithm, and there is a certain gap compared with MUM-net based algorithm and M-net based algorithm. This also shows that improving the robustness of the algorithm is our next key research work.

Fourth, in terms of tampering sensitivity, the MultiResUnet based algorithm is the worst among all comparison algorithms, and the U-net based algorithm is the best among all algorithms. This result is the same as the conclusion of [29]. Our Semi-U-net based algorithm is not as sensitive to tampering as U-net based algorithm, but it is stronger than other existing algorithms.

Overall, our Semi-U-Net-based algorithm not only achieves the goal of high efficiency and light weight, but also does not sacrifice authentication performance.

In fact, robustness and tampering sensitivity are often contradictory: stronger robustness often means weaker tampering sensitivity, and increased tampering sensitivity often means weaker robustness. Therefore, in actual integrity authentication, the setting of the threshold of the normalized hamming distance requires comprehensive consideration of the above two aspects.

V. CONCLUSION AND FUTURE WORK

Deep neural network is the key to achieve subject-sensitive hashing of HRRS images. But the existing deep neural network models for subject-sensitive hashing have shortcomings in terms of model complexity, computational performance, and occupied storage space. In this work, we present an efficient lightweight deep neural network, Semi-U-Net, for subject-sensitive hashing of HRRS images. Semi-U-Net combines the characteristics of perceptual hashing to construct an asymmetric network structure, and uses deep separable convolution instead of ordinary two-dimensional convolution to build a network model making the model lightweight. The experimental results show that the size of our Semi-U-Net model is only 5.38M, which is lighter than the existing deep neural network model, and achieves more efficient hash calculation (about 88.6fps).

Although our model has improved computational efficiency and model size compared with existing models that implement subject-sensitive hashing, it is still difficult for our model to be applied on mobile devices. In future research work, we will further study the simplification of deep neural networks and further improve the construction method of hash sequences to facilitate the comparison of hash sequences.

REFERENCES

[1] L. Hao, Y. Zhang, and Z. Cao, "Active cues collection and integration for building extraction with high-resolution color remote sensing imagery," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 12, no. 8, pp. 2675–2694, Aug. 2019, doi: 10.1109/JSTARS.2019.2926738.

[2] K. Liu, H. Ma, H. Ma, Z. Cai, and L. Zhang, "Building extraction from airborne LiDAR data based on min-cut and improved post-processing," *Remote Sens.*, vol. 12, no. 17, p. 2849, Sep. 2020.

[3] M. Dikmen and U. Halici, "A learning-based resegmentation method for extraction of buildings in satellite images," *IEEE Geosci. Remote Sens. Lett.*, vol. 11, no. 12, pp. 2150–2153, Dec. 2014, doi: 10.1109/LGRS.2014.2321658.

[4] S. Timilsina, J. Aryal, and J. B. Kirkpatrick, "Mapping urban tree cover changes using object-based convolution neural network (OB-CNN)," *Remote Sens.*, vol. 12, no. 18, p. 3017, Sep. 2020.

[5] M. Varin, B. Chalghaf, and G. Joanisse, "Object-based approach using very high spatial resolution 16-band WorldView-3 and LiDAR data for tree species classification in a broadleaf forest in Quebec, Canada," *Remote Sens.*, vol. 12, no. 18, p. 3092, Sep. 2020.

[6] J. Liu, Q. Feng, Y. Wang, B. Batsaikhan, J. Gong, Y. Li, C. Liu, and Y. Ma, "Urban green plastic cover mapping based on VHR remote sensing images and a deep semi-supervised learning framework," *ISPRS Int. J. Geo-Inf.*, vol. 9, no. 9, p. 527, Sep. 2020.

[7] L. Zhang, A. Li, Z. Zhang, and K. Yang, "Global and local saliency analysis for the extraction of residential areas in high-spatial-resolution remote sensing image," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 7, pp. 3750–3763, Jul. 2016, doi: 10.1109/TGRS.2016.2527044.

[8] D. Furberg, Y. Ban, and A. Nascetti, "Monitoring of urbanization and analysis of environmental impact in stockholm with sentinel-2A and SPOT-5 multispectral data," *Remote Sens.*, vol. 11, no. 20, p. 2408, Oct. 2019.

[9] T. Zhang and X. Huang, "Monitoring of urban impervious surfaces using time series of high-resolution remote sensing images in rapidly urbanized areas: A case study of Shenzhen," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 11, no. 8, pp. 2692–2708, Aug. 2018, doi: 10.1109/JSTARS.2018.2804440.

[10] X. Li, X. Sun, and Q. Liu, "Image integrity authentication scheme based on fixed point theory," *IEEE Trans. Image Process.*, vol. 24, no. 2, pp. 632–645, Feb. 2015, doi: 10.1109/TIP.2014.2372473.

[11] C. Zhang, S. Wei, S. Ji, and M. Lu, "Detecting large-scale urban land cover changes from very high resolution remote sensing images using CNN-based classification," *ISPRS Int. J. Geo-Inf.*, vol. 8, no. 4, p. 189, Apr. 2019.

[12] X.-M. Niu and Y.-H. Jiao, "An overview of perceptual hashing," *Acta Electron. Sinica*, vol. 36, no. 7, pp. 1405–1411, Jul. 2008.

[13] L. Du, A. T. S. Ho, and R. Cong, "Perceptual hashing for image authentication: A survey," *Signal Process., Image Commun.*, vol. 81, Feb. 2020, Art. no. 115713.

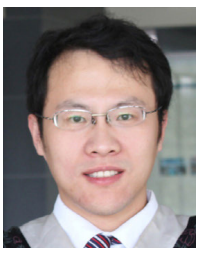
[14] C. Qin, M. Sun, and C.-C. Chang, "Perceptual hashing for color images based on hybrid extraction of structural features," *Signal Process.*, vol. 142, pp. 194–205, Jan. 2018.

[15] X. Lv and Z. J. Wang, "Perceptual image hashing based on shape contexts and local feature points," *IEEE Trans. Inf. Forensics Security*, vol. 7, no. 3, pp. 1081–1093, Jun. 2012, doi: 10.1109/TIFS.2012.2190594.

[16] R. Sun and W. Zeng, "Secure and robust image hashing via compressive sensing," *Multimedia Tools Appl.*, vol. 70, no. 3, pp. 1651–1665, Jun. 2014.

[17] H. Liu, D. Xiao, Y. Xiao, and Y. Zhang, "Robust image hashing with tampering recovery capability via low-rank and sparse representation," *Multimedia Tools Appl.*, vol. 75, no. 13, pp. 7681–7696, Jul. 2016.

[18] Z. Tang, Z. Huang, X. Zhang, and H. Lao, "Robust image hashing with multidimensional scaling," *Signal Process.*, vol. 137, pp. 240–250, Aug. 2017.
[19] M. Sajjad, I. U. Haq, J. Lloret, W. Ding, and K. Muhammad, "Robust image hashing based efficient authentication for smart industrial environment," *IEEE Trans. Ind. Informat.*, vol. 15, no. 12, pp. 6541–6550, Dec. 2019, doi: 10.1109/TII.2019.2921652.
[20] K. Ding, F. Meng, Y. Liu, N. Xu, and W. Chen, "Perceptual hashing based forensics scheme for the integrity authentication of high resolution remote sensing image," *Information*, vol. 9, no. 9, p. 229, Sep. 2018.
[21] K. Ding, Z. Yang, Y. Wang, and Y. Liu, "An improved perceptual hash algorithm based on U-Net for the authentication of high-resolution remote sensing image," *Appl. Sci.*, vol. 9, no. 15, p. 2972, Jul. 2019.
[22] X. Zhang, H. Yan, L. Zhang, and H. Wang, "High-resolution remote sensing image integrity authentication method considering both global and local features," *ISPRS Int. J. Geo-Inf.*, vol. 9, no. 4, p. 254, Apr. 2020.
[23] K. Ding, S. Chen, and F. Meng, "A novel perceptual hash algorithm for multispectral image authentication," *Algorithms*, vol. 11, no. 1, p. 6, Jan. 2018.
[24] S. Jin, H. Yao, X. Sun, and S. Zhou, "Unsupervised semantic deep hashing," *Neurocomputing*, vol. 351, no. 25, pp. 19–25, Jul. 2019.
[25] M. Zhou, X. Zeng, and A. Chen, "Deep forest hashing for image retrieval," *Pattern Recognit.*, vol. 95, pp. 114–127, Nov. 2019.
[26] G. Gu, J. Liu, Z. Li, W. Huo, and Y. Zhao, "Joint learning based deep supervised hashing for large-scale image retrieval," *Neurocomputing*, vol. 385, pp. 348–357, Apr. 2020.
[27] Y. Peng, J. Zhang, and Z. Ye, "Deep reinforcement learning for image hashing," *IEEE Trans. Multimedia*, vol. 22, no. 8, pp. 2061–2073, Aug. 2020, doi: 10.1109/TMM.2019.2951462.
[28] Q.-Y. Jiang, X. Cui, and W.-J. Li, "Deep discrete supervised hashing," *IEEE Trans. Image Process.*, vol. 27, no. 12, pp. 5996–6009, Dec. 2018, doi: 10.1109/TIP.2018.2864894.
[29] K. Ding, Y. Liu, Q. Xu, and F. Lu, "A subject-sensitive perceptual hash based on MUM-net for the integrity authentication of high resolution remote sensing images," *ISPRS Int. J. Geo-Inf.*, vol. 9, no. 8, p. 485, Aug. 2020.
[30] A. G. Howard, M. Zhu, B. Chen, and D. Kalenichenko, "MobileNets: Efficient convolutional neural networks for mobile vision applications," 2017, *arXiv:1704.0486*. [Online]. Available: https://arxiv.org/abs/1704.04861
[31] Y. Cao, B. Liu, M. Long, and J. Wang, "HashGAN: Deep learning to hash with pair conditional Wasserstein GAN," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 3270–3278.
[32] T. Yao, Y. Han, R. Wang, X. Kong, L. Yan, H. Fu, and Q. Tian, "Efficient discrete supervised hashing for large-scale cross-modal retrieval," *Neurocomputing*, vol. 385, no. 14, pp. 358–367, Apr. 2020.
[33] L. Yan, F. Zou, R. Guo, L. Gao, K. Zhou, and C. Wang, "Feature aggregating hashing for image copy detection," *World Wide Web*, vol. 19, no. 2, pp. 217–229, Mar. 2016.
[34] Z. Zhou, M. Wang, Y. Cao, and Y. Su, "CNN feature-based image copy detection with contextual hash embedding," *Mathematics*, vol. 8, no. 7, p. 1172, Jul. 2020.
[35] X. Wang, X. Zhou, Q. Zhang, B. Xu, and J. Xue, "Image alignment based perceptual image hash for content authentication," *Signal Process., Image Commun.*, vol. 80, Feb. 2020, Art. no. 115642.
[36] H. Li, Y. Wu, and M. Chen, "Adaptive fault-tolerant tracking control for discrete-time multiagent systems via reinforcement learning algorithm," *IEEE Trans. Cybern.*, vol. 51, no. 3, pp. 1163–1174, Mar. 2021, doi: 10.1109/TCYB.2020.2982168.
[37] A. B. Nassif, I. Shahin, I. Attili, M. Azzeh, and K. Shaalan, "Speech recognition using deep neural networks: A systematic review," *IEEE Access*, vol. 7, pp. 19143–19165, 2019, doi: 10.1109/ACCESS.2019.2896880.
[38] L. Claussmann, M. Revilloud, D. Gruyer, and S. Glaser, "A review of motion planning for highway autonomous driving," *IEEE Trans. Intell. Transp. Syst.*, vol. 21, no. 5, pp. 1826–1848, May 2020, doi: 10.1109/TITS.2019.2913998.
[39] P. Du, Y. Pan, H. Li, and H.-K. Lam, "Nonsingular finite-time event-triggered fuzzy control for large-scale nonlinear systems," *IEEE Trans. Fuzzy Syst.*, early access, May 6, 2020, doi: 10.1109/TFUZZ.2020.2992632.
[40] Y. Pan, P. Du, H. Xue, and H.-K. Lam, "Singularity-free fixed-time fuzzy control for robotic systems with user-defined performance," *IEEE Trans. Fuzzy Syst.*, early access, Jun. 3, 2020, doi: 10.1109/TFUZZ.2020.2999746.
[41] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *Proc. 18th Int. Conf. Med. Image Comput. Comput.-Assist. Intervent.*, Oct. 2015, pp. 234–241.
[42] N. Ibtehaz and M. S. Rahman, "MultiResUNet: Rethinking the U-Net architecture for multimodal biomedical image segmentation," *Neural Netw.*, vol. 121, pp. 74–87, Jan. 2020.
[43] J. Deng, Z. Zhong, H. Huang, Y. Lan, Y. Han, and Y. Zhang, "Lightweight semantic segmentation network for real-time weed mapping using unmanned aerial vehicles," *Appl. Sci.*, vol. 10, no. 20, p. 7132, Oct. 2020.
[44] X. Hu, H. Li, X. Li, and C. Wang, "MobileNet-SSD MicroScope using adaptive error correction algorithm: Real-time detection of license plates on mobile devices," *IET Intell. Transp. Syst.*, vol. 14, no. 2, pp. 110–118, Feb. 2020.
[45] B. Zhang, Y. Zhang, and S. Wang, "A lightweight and discriminative model for remote sensing scene classification with multidilation pooling module," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 12, no. 8, pp. 2636–2653, Aug. 2019.
[46] L. Chen, Z. Wei, and Y. Xu, "A lightweight spectral–spatial feature extraction and fusion network for hyperspectral image classification," *Remote Sens.*, vol. 12, no. 9, p. 1395, Apr. 2020.
[47] H. Dong, L. Zhang, and B. Zou, "PolSAR image classification with lightweight 3D convolutional networks," *Remote Sens.*, vol. 12, no. 3, p. 396, Jan. 2020.
[48] Y. Wang, C. Chen, M. Ding, and J. Li, "Real-time dense semantic labeling with dual-path framework for high-resolution remote sensing image," *Remote Sens.*, vol. 11, no. 24, p. 3020, Dec. 2019.
[49] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *J. Mach. Learn. Res.*, vol. 15, no. 1, pp. 1929–1958, 2014.
[50] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 3431–3440.
[51] D. Molchanov, A. Ashukha, and D. Vetrov, "Variational dropout sparsifies deep neural networks," in *Proc. 34th Int. Conf. Mach. Learn. (ICML)*, Jul. 2017, pp. 2498–2507.
[52] V. Adiga and J. Sivaswamy, "FPD-M-net: Fingerprint image denoising and inpainting using m-net based convolutional neural networks," in *Inpainting and Denoising Challenges*. Heidelberg, Germany: Springer, 2019, pp. 51–61.
[53] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 2, pp. 318–327, Feb. 2020, doi: 10.1109/TPAMI.2018.2858826.
[54] S. P. Ji and S. Y. Wei, "Building extraction via convolutional neural networks from an open remote sensing building dataset," *Acta Geodaetica et Cartographica Sinica*, vol. 48, no. 4, pp. 448–459, 2020.
[55] Y. Cheng, S. Jin, M. Wang, Y. Zhu, and Z. Dong, "Image mosaicking approach for a double-camera system in the GaoFen2 optical remote sensing satellite based on the big virtual camera," *Sensors*, vol. 17, no. 6, p. 1441, Jun. 2017.
[56] G.-S. Xia, X. Bai, J. Ding, Z. Zhu, S. Belongie, J. Luo, M. Datcu, M. Pelillo, and L. Zhang, "DOTA: A large-scale dataset for object detection in aerial images," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2018, pp. 3974–3983.



KAIMENG DING received the B.E. degree in computer science and technology from Jiangsu Ocean University, Lianyungang, China, in 2009, and the Ph.D. degree in geographic information system (GIS) from Nanjing Normal University, Nanjing, China, in 2015. He is currently an Associate Professor with the School of Networks and Telecommunications Engineering, Jinling Institute of Technology. His current research interests include geographic data security and deep-learning application.

SHOUBAO SU received the B.S., M.S., and Ph.D. degrees in computer science and technology from Anhui University, in 1986, 2004, and 2009, respectively. He is currently a Professor with the Jiangsu Key Laboratory of Data Science and Smart Software. His current research interests include swarm intelligence algorithms, deep neural networks, and information security.

NAN XU received the B.S. degree in computer science and technology from Henan Normal University, in 1996, the M.D. degree in computer science and technology from Nanjing Normal University, in 2005, and the Ph.D. degree in computer science and technology from the Nanjing University of Science and Technology, in 2012. She is currently an Associate Professor with the School of Intelligent Science and Control Engineering, Jinling Institute of Technology. Her current research interests include artificial intelligence applications and data security.

TINGTING JIANG received the B.E. degree in environmental science and engineering from Nanjing Normal University, Nanjing, China, in 2012. She is currently an Engineer with Ericsson Communications Company Ltd., Nanjing Branch, China. Her current research interests include the protection technology of digital circuit diagrams, the usability evaluation of safety technology, and the packaging defect detection technology based on artificial intelligence technology.