International Journal of Geo-Information

MDPI

Article

A Subject-Sensitive Perceptual Hash Based on MUM-Net for the Integrity Authentication of High Resolution Remote Sensing Images

Kaimeng Ding [1,2,*], Yueming Liu [2], Qin Xu [1,*] and Fuqiang Lu [3]

1 School of Networks and Tele-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; liuym@lreis.ac.cn
3 School of Computer Science and Information Engineering, Changzhou Institute of Technology, Changzhou 213022, China; lufq@czu.cn
* Correspondence: dkm@jit.edu.cn (K.D.); missxuqin@jit.edu.cn (Q.X.); Tel.: +86-181-6809-2159 (K.D.); +86-181-6809-2195 (Q.X.)

Received: 3 July 2020; Accepted: 9 August 2020; Published: 11 August 2020

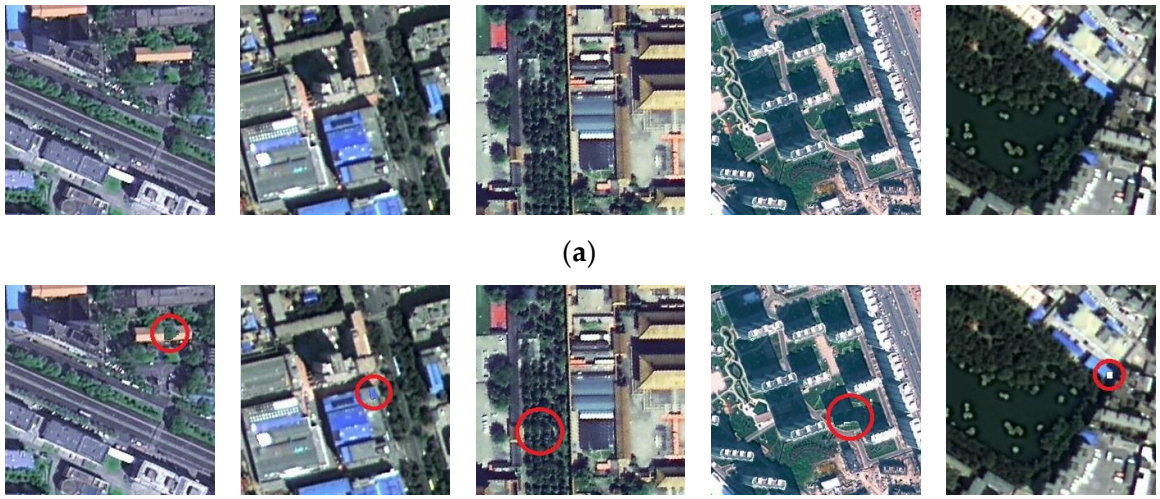
Abstract: Data security technology is of great significance to the application of high resolution remote sensing image (HRRS) images. As an important data security technology, perceptual hash overcomes the shortcomings of cryptographic hashing that is not robust and can achieve integrity authentication of HRRS images based on perceptual content. However, the existing perceptual hash does not take into account whether the user focuses on certain types of information of the HRRS image. In this paper, we introduce the concept of subject-sensitive perceptual hash, which can be seen as a special case of conventional perceptual hash, for the integrity authentication of HRRS image. To achieve subject-sensitive perceptual hash, we propose a new deep convolutional neural network architecture, named MUM-Net, for extracting robust features of HRRS images. MUM-Net is the core of perceptual hash algorithm, and it uses focal loss as the loss function to overcome the imbalance between the positive and negative samples in the training samples. The robust features extracted by MUM-Net are further compressed and encoded to obtain the perceptual hash sequence of HRRS image. Experiments show that our algorithm has higher tamper sensitivity to subject-related malicious tampering, and the robustness is improved by about 10% compared to the existing U-net-based algorithm; compared to other deep learning-based algorithms, this algorithm achieves a better balance between robustness and tampering sensitivity, and has better overall performance.

Keywords: perceptual hash; integrity authentication; subject-sensitive; HRRS image; deep learning

1. Introduction

As important geographic data, high-resolution remote sensing (HRRS) images are widely used in geographic information extraction [1,2], surveying and mapping [3,4], Earth resource surveys [5], geological disaster investigation and rescue [6], land use [7], military reconnaissance [8] and other fields [9,10]. At the same time, the rapid development of modern network technology and image processing technology makes HRRS images vulnerable to various unintentional or intentional tampering attacks during their transmission, storage, and use. Since high precision and confidentiality are important characteristics of HRRS images, the use value of the HRRS images will be greatly reduced if people are not sure whether the HRRS images have been tampered with. Even more serious, the tampered HRRS image may even lose application value.

Figure 1 shows several examples of tampered HRRS images. For these tampered HRRS images, it is difficult to determine whether they have been tampered even if users compare them with the original images. Moreover, it is often impossible to compare original images in reality. In this case, data security technologies, especially integrity authentication technologies for HRRS images, have gradually attracted attention.



(**a**)

(**b**)

Figure 1. Examples of tampered HRRS images: (**a**) Original HRRS images, (**b**) tampered images, and the tampering operations (from left to right) are: a building is modified, a building is added, a tree is added, a building is deleted, and random malicious cut-out is performed.

Authentication technology protects data by verifying its integrity and authenticity. Among the current major authentication technologies, perceptual hash [11,12] has the characteristics of being able to perceive the content of multimedia data. Perceptual hash, also known as perceptual hashing, can map the perceptual content of multimedia data into a digital digest called perceptual hash sequence or perceptual hash value. Compared with other authentication technologies (such as cryptographic hash, digital signatures, fragile watermarking, etc.), perceptual hash can better realize the authentication of HRRS image because of its robustness. Although the existing perceptual hash algorithms for HRRS image in [13–15] can solve the problem of HRRS image authentication to a certain extent, the authentication problem of HRRS images still cannot be completely solved.

The key to executing image authentication based on perceptual hash is to extract the perceptual content of HRRS image. At present, the following features are generally used as perceptual contents of images in mainstream perceptual hash algorithms [16]: the coefficients of transform domains [17,18], the local feature points [19–21], features after image dimensionality reduction [22–24], statistics features [25,26], and other features (such as [27]).

Actually, for specific applications of HRRS images, users often have different authentication requirements. For example, users who take hydrology as the main research object often pay more attention to rivers and lakes in remote sensing images; users who study outdoor navigation often pay more attention to roads and bridges in the images; users who study moving target extraction tend to pay attention to aircraft, ships, large vehicles, etc. Therefore, different users' requirements for integrity authentication are often related to certain subjects for different application fields.

Based on this view, we believe that the material information with higher value to users should be more strictly authenticated. In other words, the tampering that users are concerned about is often related to a certain subject. Here, we take the building information as an example to compare the subject related tampering and subject unrelated tampering, as shown in Figure 2.

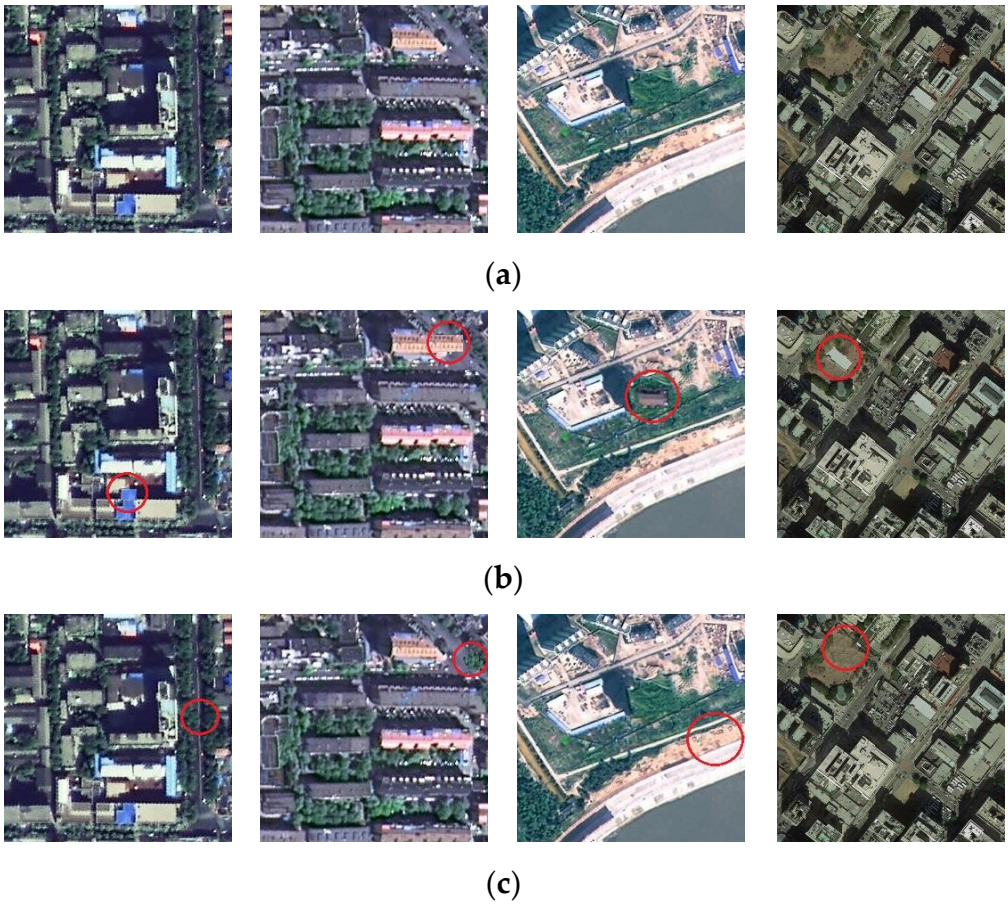


Figure 2. Example of subject related tampering and subject unrelated tampering: (**a**) Original HRRS images, (**b**) subject-related tampering, (**c**) subject unrelated tampering.

Inspired by the fact that users pay different attention to different content in HRRS images, we introduce the concept of "subject-sensitive perceptual hash" to meet this "subject-biased" integrity authentication for HRRS images. Subject-sensitive perceptual hash is special case of traditional perceptual hash. For example, if the user primarily uses building information in the HRRS image, then any changes to the building's effective information should be detected by the subject-sensitive perceptual hash algorithm, while if other contents is changed, such as forests and lakes, there is no need to perform too much strict authentication.

However, it would be very difficult to execute subject-sensitive perceptual hash algorithms based on traditional remote sensing image processing technology which does not have the ability to learn feature extraction rules from samples. Fortunately, the rise of deep learning provides a feasible way for the implementation of subject-sensitive perceptual hash. Compared with traditional methods, deep learning has excellent feature expression capabilities and can mine the essential information hidden inside the data [28–32]. Deep learning-based methods can automatically learn more essential features from training samples, reduce the complexity of artificial design features, and then enhance the performance of perceptual hash authentication algorithms. For perceptual hash, including subject-sensitive perceptual hash, it is not that the more features extracted by the model, the better, nor the more complex features extracted, the better, but the extracted features should be as balanced as possible between robustness and tampering sensitivity. In recent years, learning-based perceptual hash algorithms has also been gradually studied [13,33], but their original intention was not subject-sensitive perceptual hash.

To achieve subject-sensitive perceptual hash, in this paper we propose a new deep convolutional neural network, called multi-scale U-shape chained M-shape convolution network (MUM-Net), for the

robust feature extraction of the perceptual hash algorithm. To verify the effectiveness of our proposed method, we not only compare our proposed perceptual hash algorithm based on MUM-Net with the traditional perceptual hash algorithm in the experiments, but also compare MUM-Net with other convolutional neural networks such as U-net [34], M-net [35] and MultiResUnet [36].

Overall, the contributions of this work are as follows:

(1) We introduce the concept of subject-sensitive perceptual hash, which can be considered as a special case of the traditional perceptual hash.
(2) We propose a method for constructing training sample sets to achieve subject-sensitive perceptual hash. This method can effectively use existing data sets and adjust the algorithm performance according to actual needs.
(3) We propose a convolutional neural network (CNN) architecture named MUM-Net for the subject-sensitive feature extraction of HRRS images. This CNN architecture is the key to implement subject-sensitive perceptual hash.

The composition of this paper is as follows: The current related works are described in Section 2. Section 3 discusses the details of our proposed subject-sensitive perceptual hash algorithm and MUM-Net. The details of the experimental setup and the result are presented respectively in Sections 4 and 5. A discussion is presented in Section 6, and the conclusions are drawn in Section 7.

2. Related Work

With the rapid development of satellite remote sensing technology and unmanned aerial vehicle (UAV) technology, the resolution of HRRS images is getting higher and higher. For example, the spatial resolution of the GeoEye satellite reaches 0.41 m, and the WorldView-3 satellite has an imaging capability of 0.3 m resolution. For HRRS image, the integrity authentication technology should ensure whether the effective content information carried by the images has changed. If the content of the HRRS image has not changed, and only the carrier has changed, it cannot be considered that the HRRS image has been tampered. For example, if the HRRS image is format-converted, its binary level representation may have changed dramatically, but the content carried has not changed. The cryptographic authentication algorithm aims to perform binary level authentication on the data, which is suitable for text data, but not suitable for remote sensing images.

Perceptual hash inherits most of the characteristics of cryptographic hash functions, such as unidirectionality, anti-collision, and summary, and can convert any length of input information into a short output sequence. At the same time, perceptual hash also has the characteristics that cryptographic hash functions do not have, that is, perceptual hash is robust.

The general steps of the perceptual hash algorithm include: (1) Image preprocessing to make the image more convenient for feature extraction; (2) Feature extraction, extracting the perceptual features of the image through SIFT (Scale-invariant feature transform), matrix decomposition, DWT (Discrete Wavelet Transform) and other methods; (3) Feature quantization to remove the features redundancy; (4) In the feature encoding stage, compression, encryption and other operations are performed on the features to obtain the final perceptual hash sequence. Among them, feature extraction is a key step, which has an important influence on the distinguishability, robustness, and summary of the algorithm.

In the related research for HRRS images, a perceptual hash scheme based on U-net is proposed in [13] for the authentication of HRRS images, which may be the first attempt to explore the application of deep neural network to perceptual hash of HRRS image. However, this method is an improvement on the traditional perceptual hash algorithm and cannot be directly used to achieve subject-sensitive perceptual hash. In [14], a perceptual hash technology considering both global and local features is proposed, which combine Zernike moments and Features from Accelerated Segment Test (FAST) feature descriptors. In [15], a perceptual hash algorithm based on a Gabor filter bank is proposed. Like the algorithm in [14], this algorithm is also based on traditional image processing methods and cannot achieve subject-sensitive perceptual hash.

The abovementioned perceptual hash algorithms related to remote sensing images [13–15] and perceptual hash algorithms for ordinary images [16–27] have the following problems:

(1) Most of the algorithms mentioned above use traditional feature extraction methods, which are artificially designed visual features. However, the "semantic gap" problem indicates that the content of HRRS images cannot be fully represented by visual features alone. For example, the perceptual hash authentication algorithm based on feature points can easily use "false feature points" caused by light and fog as the perceptual characteristics of remote sensing images, which greatly reduces the authentication performance of perceptual hash algorithms.
(2) For specific applications of HRRS images, the robustness of existing perceptual hash authentication algorithms often has some shortcomings, as the artificially designed features cannot express more abstract high-dimensional features of the underlying features of the HRRS image, that is, the essential features of the remote sensing image in the application cannot be tapped, and the certification requirements of the HRRS image in a complex environment cannot be met.
(3) The focus of HRRS image content that different types of users pay attention to is often different, which means that the perceptual hash algorithm should try to identify whether the target that the user is interested in has been tampered, while existing perceptual hash algorithms do not take this into account. For example, if the user mainly uses the information of buildings or roads in the HRRS images, the authentication algorithm should pay more attention to the addition, deletion, or change of buildings or roads in the images, and should appropriately maintain a certain degree of robustness to other categories of targets such as grasses and ponds.

In response to the above problems, we introduce the concept of "subject-sensitive perceptual hash" to achieve integrity authentication with subjectivity emphasis. We consider the subject-sensitive perceptual hash to be a special case of conventional perceptual hash. In the extreme case, if the user pays attention to the changes of all objects in the image, the subject-sensitive perceptual hash will degenerate into the conventional perceptual hash.

In this paper, the subject-sensitive perceptual hash is defined as follows: The 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. Subject-sensitive perceptual hash should be able to detect subject-related tampering with higher sensitivity. Let's take buildings as an example of a sensitive subject: if the building in the image is added, deleted, or changed, the perceptual hash sequence should change drastically. However, this does not mean that subject-sensitive perceptual hash cannot detect subject-unrelated tampering, but that it can still detect subject unrelated tampering, only that the sensitivity is reduced. For example, if there is a slight change in the grass or river in the image, the change of the perceptual hash sequence should be very small.

Deep learning provides an excellent technical way to implement subject-sensitive perceptual hash. For visual media data such as images, the essence of deep learning is to simulate human visual perception, that is, to form abstract deep features by combining shallow features and discovering hidden features in the data. Therefore, deep learning can simulate the "subject-bias" of HRRS users, which is not easy to do with traditional image processing methods.

In the existing research on the perceptual hash based on deep learning for HRRS image authentication, the U-net-based perceptual hash algorithm proposed in [13] has applied deep learning methods to the authentication of HRRS images. However, the algorithm in [13] still has the following problems that make it cannot fully achieve "subject-sensitive integrity authentication".

The algorithm in [13] aims to improve the robustness of the algorithm through deep learning, and experiments have shown that the robustness of the algorithm has indeed improved. However, the robustness of the algorithm is still in the traditional sense, and it does not combine the user's attention, which is different from the subject-sensitive hash introduced in this paper. Therefore, it is still not robust to some pixel-level operations that do not affect user use. For example, if the user

mainly uses the building information in the HRRS image, then if the user modifies the fine bush or vehicle information in the HRRS image, it will be considered as a malicious tampering operation.

The algorithm in [13] improves the existing perceptual hash algorithm for HRRS image, and uses U-net to extract the features of HRRS images. However, no new deep network has been designed, but the traditional U-net network has been slightly modified. As a result, the improvement of the robustness of the algorithm is limited.

In this paper, we propose a new convolutional neural network architecture, named MUM-Net, referring to U-net [34] to extract the robust features of HRRS images. U-net, which was first used for medical image segmentation, is an improved fully convolutional neural (FCN) network [37]. U-Net and its various improved networks have an encoder-decoder structure, which combines the low-dimensional and high-dimensional features of the convolutional network and ensures the consistency of the input and output resolution with fine precision. The MUM-Net proposed is the core of our subject-sensitive perceptual hash.

3. Developed Method

In this section, we will first show the overall framework of the subject-sensitive perceptual hash algorithm, and then focus on the details of MUM-Net. Finally, the HRRS image authentication process, which is the application process of subject-sensitive perceptual hash, will be discussed.

3.1. Overview of Developed Subject-Sensitive Perceptual Hash Algorithm

Compared with conventional perceptual hash, subject-sensitive perceptual hash is characterized by using the trained MUM-Net model to extract the perceptual features of HRRS images, rather than using traditional image feature extraction techniques. As shown in Figure 3, the proposed subject-sensitive perceptual hash consists of three steps: The first step is the pre-processing of HRRS image to perform grid division, normalization, and band fusion. In the second step, the perceptual feature of the HRRS image would be extracted based on our proposed MUM-Net. The final step is the compression encoding, which will get the perceptual hash sequence of the HRRS image.

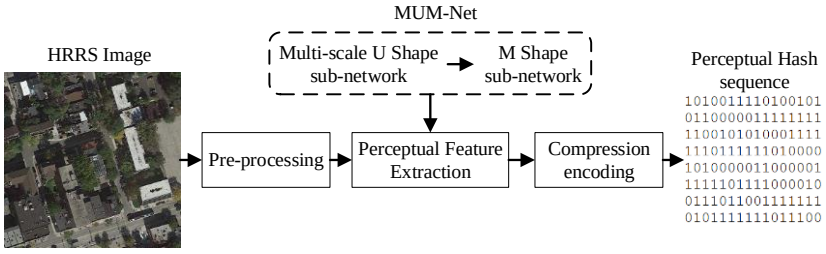


Figure 3. Overview of the proposed perceptual hash algorithm.

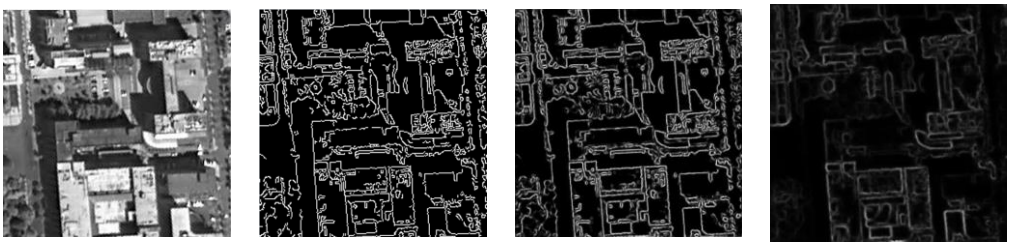
Taking into account the characteristics of the massive data of HRRS images, the algorithm in this paper uses similar methods of other perceptual hash algorithms for remote sensing image [13–15] to divide the HRRS images into $W \times H$ grids, and then generates perceptual hash sequence for each grid cell separately. Then, bilinear interpolation is used to normalize each grid cell to $m \times m$ pixel size. In this paper, $m = 256$, which is the size of the input image of MUM-Net. The normalized grid cell is denoted as R_{ij}.

After the HRRS image is preprocessed, the perceptual features of the grid cell R_{ij} are extracted through MUM-Net, and the extracted robust features are denoted as F_{ij}.

Edge information takes an important position in the effective use of HRRS images, and edge features can identify subtle tampering of images with high accuracy, which is more suitable for the perceptual hash authentication algorithm of remote sensing images. In addition, edge features can better meet the accuracy requirements of remote sensing images [13,15]. Therefore, the algorithm in this paper also uses the edge feature as the basis to construct the perceptual hash sequence of the HRRS

image. That is to say, the MUM-Net trained based on our constructed dataset will extract the robust edge features of the grid cell.

To illustrate the features extracted by MUM-Net more vividly, we compare the intermediate results extracted by MUM-Net with the Canny results and the method in [13], as shown in Figure 4. Figure 4a shows the grid unit after preprocessing, Figure 4b is the extraction result of Canny, Figure 4c is the extraction result of the method in [13], and Figure 4d is the extraction result of MUM-Net. The training samples used by MUM-Net will be discussed in Section 4.1, and the method in [13] uses its own training data set.



(**a**) (**b**) (**c**) (**d**)

Figure 4. Comparison of different feature extraction methods: (**a**) Grid cell after preprocessing, (**b**) edge detected by Canny operator, (**c**) edge detected by the method in [13], (**d**) robust edge detected by MUM-Net.

Obviously, Canny extracts too many edge features that contain many useless features. If the perceptual hash sequence is constructed based on the results of Canny extraction, the robustness of the algorithm will be relatively poor. The edge features extracted by the method in [13] obviously lack a lot of subtle edge features, which greatly improves the robustness of the algorithm. However, there is a certain gap between this method and the subject sensitive perceptual hash in this paper. It can be seen in Figure 4d that the noise of the edge feature extracted by MUM-Net is greatly reduced, while the edge feature of the subject-related features such as buildings is clearly visible, which meets the requirements of the algorithm in this paper.

In the compression encoding stage, the robust edge features F_{ij} of each grid cell will be digested to construct the perceptual hash sequence which is mainly executed through principal component analysis (PCA) and string encryption.

The perceptual features F_{ij} extracted by MUM-Net is essentially a two-dimensional matrix of pixel gray values. After PCA decomposition, the first few principal components contain most of the content information of F_{ij}. Therefore, the algorithm selects the principal component as the digested perceptual feature of the grid cell. The selected principal components are then binarized to a 0-1 sequence. Then the advanced encryption standard (AES) algorithm is used for encryption processing to obtain the perceptual hash sequence of the grid cell, which is denoted as PH_{ij}. The perceptual hash sequence PH_{ij} of all grid cells is concatenated to obtain the final perceptual hash sequence of the HRRS image, which is recorded as PH.

3.2. Architecture of MUM-Net

MUM-Net is composed of a multi-scale U-shape convolution network (referred to as MU sub-net) and an improved M-net [35] (referred to as M sub-net) as demonstrated in Figure 5, and the size of its input image is 256×256. The two sub-networks are essentially improvements of U-net: the former, namely multi-scale U shape sub-network (details will be discussed in Section 3.2.1), mainly extracts the edge features of features as rich as possible; the latter; M shape sub-network is our improvement based on M-net (details will be discussed in Section 3.2.2), and is used to obtain more robust edge features.

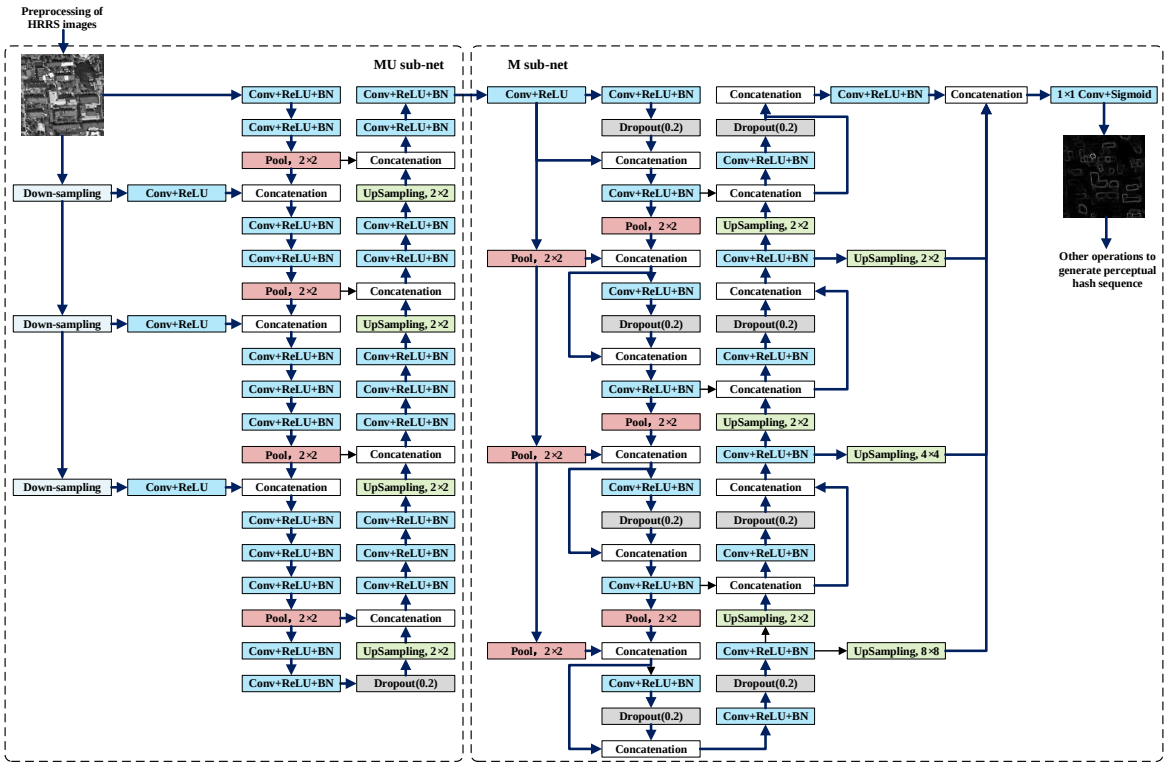


Figure 5. Detailed structure of our MUM-Net architecture. (Conv represents 3 × 3 convolution operation, ReLU (rectified linear unit) is the activation function, BN is the batch normalization layer, Concatenation is the feature fusion operation, Down-sampling is down sampling, UpSampling is up sampling, and Dropout is used to prevent overfitting.).

Generally speaking, the deeper the network level, the better the non-linear expression ability, the more complex transformations can be learned, and the more complex feature inputs can be fitted [38], that is, it can improve the network's ability to extract image features. In this way, this dual-network structure combining multi-scale U-net and improved M-net can not only extract more robust features of HRRS image, but also indirectly increase the model depth and improve the network's feature fitting ability.

We are not the first to propose this structure similar to the combination of two U-nets. In [39], a W-shaped architecture was proposed for fully unsupervised image segmentation, which ties two U-net like architecture together into a single one. The first U-net encodes an input image into a k-way soft segmentation, and the second one reverses this process to reconstruct the image. In [40], a dual U-Net architecture was proposed for nucleus segmentation. The proposed network adopts a pair of up-sampling paths sharing the down-sampling path to refine the segmentation and is capable of extracting accurate nuclear segmentation results. In [41], a network chaining a U-Net with a residual U-Net is proposed for retinal blood vessels segmentation.

Unlike the existing research on image segmentation, MUM-Net is not simply to extract the information of HRRS image, but to make the extracted features to meet the requirements of perceptual hash. For MUM-Net, the following aspects should be emphasized: On the one hand, the model should extract as many subject-sensitive features as possible, so that the algorithm's tampering sensitivity will be enhanced. On the other hand, the features extracted should be as robust as possible, that is to say, false features (such as edges caused by light or fog) and noise-like features (such as edge generated by two or three pixel mutations) should be eliminated.

Compared with W-net [39], in which the two U-net-like architecture are used as encoders and decoders respectively, the two U-net-like architecture in our model, namely MU sub-net and M sub-net respectively, assume the two tasks of multi-scale feature extraction and feature denoising. Unlike Dual

U-net [40], in which the two U-net like networks share the down-sampling path, the multi-scale U shape sub-network and M shape sub-network in our model are independent from each other, and a sigmod activation function is added between them. The network model in [41] is also quite different from our MUM-Net.

3.2.1. MU Sub-Net

The architecture of MU sub-net, which encompasses an encoder–decoder architectural ended at an active layer, is illustrated in the left half of Figure 5. The biggest difference between MU sub-net and original U-net is that MU sub-net adds multi-scale inputs to extract richer content features. In the original U-net, pooling operation was used to down-sample. However, pooling operation often causes images to lose a lot of valuable information, especially discarded position information, which results in loss of accurate spatial relative relationships between image parts, resulting in insufficient feature information. Multi-scale input can compensate for the expected position information lost in the pooling operation [42–45]. For example, ICNet (Image Cascade Network) [42] effectively uses the semantic information of low-resolution images and the detailed information of high-resolution images through multi-scale input to achieve real-time semantic segmentation of images.

In MU sub-net, we add multi-scale input on the basis of the original U-net to extract edge features of HRRS image. It includes one encoder section (also called contraction path) to capture context information and one symmetric decoder section (also called extension path) that supports precise localization. The encoder section consists of repeated 3×3 convolutional layers, nonlinear ReLU activation function, and pooling layers blocks. Among them, the pooling layer implements down-sampling, and MaxPooling is used in our experiment. In the contraction path, there are four pooling layers. The detailed process is shown in the left half of Figure 5. Unlike the original U-net, MU sub-net adds multi-scale data input, that is, four different resolutions, such as the original image, two times down-sampled image, four times down-sampled image, and eight times down-sampled image as input data. In addition to the original image, the image of each scale is subjected to a convolutional layer and then fused with the results of the corresponding scale in the original U-net. In this way, the global features lost in the pooling operation can be compensated by low-scale inputs, to improve the impact of the pooling operation.

For the decoder section, the layers are organized in reverse order to the encoder section. Each step first uses deconvolution (2×2 upper convolution) to halve the number of feature channels. After deconvolution, the result of the deconvolution is stitched with the feature map of the corresponding step in the contraction path, and the stitched image is connected with several 3×3 convolution layers and a ReLU activation function. At the end of the expansion path, there is a 1×1 convolution. Unlike U-net, which ends with a sigmod activation function, MU sub-net ends with ReLU, and the output will be used as input to the M sub-net.

3.2.2. M Sub-Net

The M-shape convolution network (referred to as M sub-net) is an improvement based on M-net. It is mainly responsible for the denoising of the extracted features, which is similar to M-net's completion of fingerprints [35]. The M sub-net takes the output of MU sub-net as input, and its output is the perceptual feature of the grid cell of HRRS image, as illustrated in the right half of Figure 5.

M-net itself is an improvement on U-net. In addition to the encoder and decoder paths in U-net, two side paths are added to provide deep supervision. The structure of M sub-net refers to the original M-net, and has been improved in the following aspects:

(1) A new loss function is used in M sub-net, that is, binary focal loss is used to replace the binary crossentropy used in the original M-net. This is because the training datasets we establish in Section 4.1 have the problem of an imbalance in the proportion of positive and negative samples, binary focal loss is used here to overcome this problem.

(2) The input data of the M sub-net is the output of the previous sub-network, and the input of the original M-net is the fingerprint image, which determines that our M sub-network's role is more for denoising instead of image segmentation.

3.2.3. Loss Function

The subject-sensitive edge features extracted by our model are essentially a binary classification of pixels in HRRS images. For training samples, the background and the internal area of the subject object often occupy most of the space, and the pixels corresponding to the subject-sensitive edge features directly related to authentication are often a few pixels. Therefore, MUM-Net uses Focal Loss [46–48] as the loss function to overcome this sample imbalance problem.

FL (focal loss) was originally proposed in [46] to solve the imbalance of positive and negative samples in the target detection problem. The loss function reduces the weight of a large number of simple negative samples in the training process, which mines effective information in difficult samples. The definition of FL is as follows:

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

$$p_t = \begin{cases} p, \text{ if } y = 1 \\ 1-p, \text{ otherwise} \end{cases} \tag{2}$$

where p is estimated probability that ranges from [0, 1], γ is tunable focusing parameter that controls the loss and its value is greater than 0, and y is the ground truth class either 0 or 1. Both p and γ are used to optimize the performance of the models.

For robust edge features, each pixel in the image is either an edge feature point or not, which is essentially a binary classification problem. As the core part of the perceptual hash algorithm, the last layer of MUM-Net is the sigmoid activation function, which maps the pixels to the probability of two classifications, a number between 0-1. According to the present analysis, we use α-balanced variant of the FL [48] as loss function:

$$\mathrm{FL}(p_t) = -\alpha_t(1-p_t)^{\gamma}\log(p_t) \tag{3}$$

where α is a value ranges from [0, 1], and $\gamma \geq 0$. As the purpose of γ is to controls the loss, the larger the value of γ, the larger the loss for badly learned instances.

In practical applications, the values of α and γ are determined according to the information of the ground features in the training sample of the HRRS image. In the experiments, we determined through experiments that $\alpha = 0.25$ and $\gamma = 2$ are the optimal settings, which makes the perceptual hash algorithm relatively optimal in terms of robustness and tampering sensitivity.

3.3. Integrity Authentication Process

The integrity authentication process of HRRS images through subject-sensitive perceptual hash is shown in Figure 6.

The process of HRRS image integrity authentication is as follows: First, the HRRS image to be authenticated is preprocessed and divided into grid cells; then the same model as the sender is used to extract the perceptual features of the grid cells; the extracted perceptual features are processed to obtain the perceptual hash sequence of the grid cell, and perceptual hash sequence of the HRRS image is obtained at last. By comparing the perceptual hash sequence of the HRRS image to be authenticated with the perceptual hash sequence of the original HRRS image, the authentication of the HRRS image to be authenticated can be achieved.

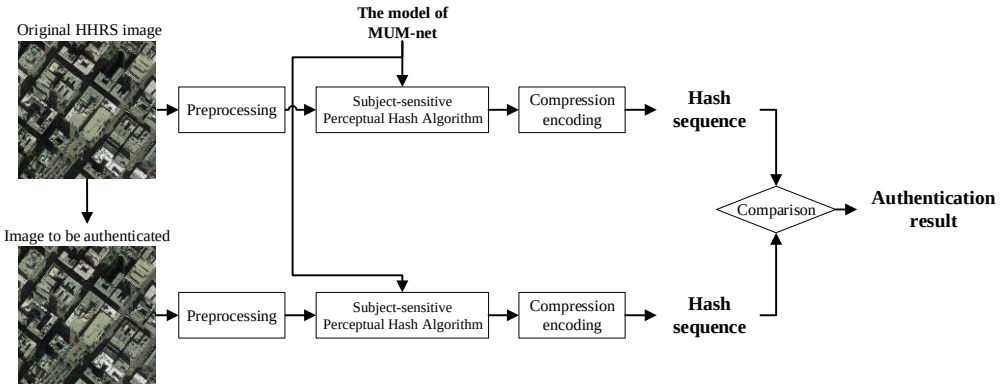


Figure 6. Integrity authentication process

For the matching process of perceptual hash sequence, we adopt the "normalized Hamming distance" [49] shown below:

$$Dis = \left(\sum_{i=1}^{StringLength} \left| hash1(i) - hash2(i) \right| \right) / StringLength \quad (4)$$

where $hash_1$ and $hash_2$ are perceptual hash sequences of the HRRS images, respectively. If the normalized Hamming distance between two perceptual hash sequences is greater than the preset threshold Dis, it means that the feature information in the corresponding area has changed significantly.

4. Experimental Setup

In this section, we will introduce training data sets, implementation details and evaluation parameters. Since different evaluation metric of subject-sensitive perceptual hash requires different testing data, the testing data sets of our algorithm will be discussed together with the experimental results in Section 5.

4.1. Training Datasets

In view of the characteristics of subject-sensitive perceptual hash, combined with the existing training sample data set for target detection of remote sensing image and the training data production method in [13], we propose a MUM-Net-oriented training sample set construction method, which can effectively use existing data sets and adjust the algorithm performance according to actual needs. This is because, instead of simply extracting the boundary information of the target features in the HRRS image, MUM-Net extracts the subject-sensitive features to generate the perceptual hash sequence. The extracted features must be able to detect whether the perceptual content of the HRRS image has changed, especially subject-related features, such as adding subject objects, deleting subject objects, changing subject objects, etc.

The implementation process is divided into three steps:

Step 1. On the basis of the existing HRRS image data set, construct a training sample for this algorithm, called "subject edge feature training sample set", and record it as TS_1. Here, we use the WHU building dataset [50] as the basis to generate training samples for our algorithm.

The main operations include: downsampling the original HRRS image from WHU building dataset and its corresponding label image to make them conform to the input size of MUM-Net that is 256×256; using the canny operator to extract the edge features of the original label image to obtain label images for MUM-Net. Figure 7 shows a set of examples, where Figure 7a shows original HRRS images from WHU building dataset, Figure 7b shows corresponding label images from WHU building dataset, and Figure 7c shows label images for MUM-Net.

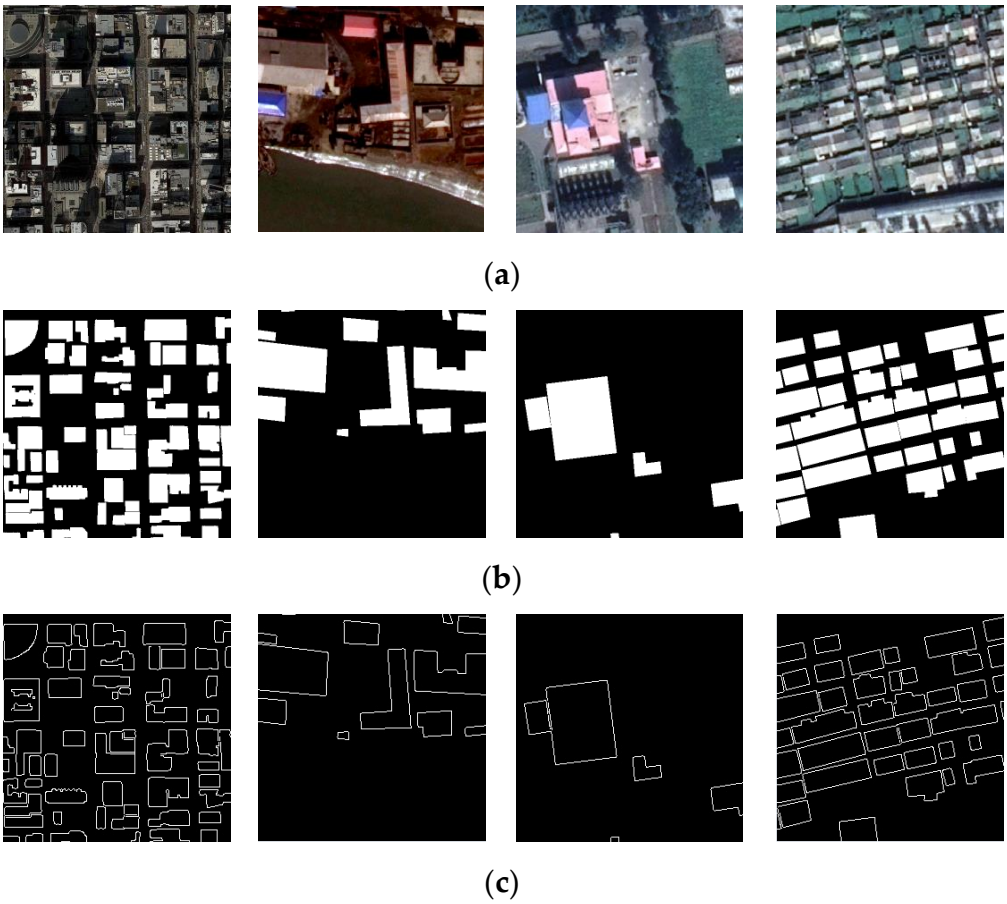


Figure 7. Example of subject edge feature training sample for MUM-Net: (**a**) Original HRRS images from WHU building dataset, (**b**) corresponding label images from WHU building dataset, (**c**) label images for MUM-Net.

Step 2. Use the method in [13] to generate a robust edge feature sample set, denoted as TS_2. The size of each image and the corresponding label are 256 × 256 pixels.

The method in [13] can be briefly summarized as follows: first, the edge features of the HRRS image are extracted using the canny operator, then the false features are manually deleted, and finally some edge features that are not detected are manually added. Figure 8 shows a set of examples. Among them, Figure 8a shows original HRRS images, and Figure 8b shows the label image produced by the method in [13].

Step 3. Select the appropriate number of training samples from TS_1 and TS_2 to construct a training sample set for MUM-Net, and record it as TS, as follows:

$$TS = TS_1' \cup TS_2' \tag{5}$$

$$TS_1' \subseteq TS_1 \tag{6}$$

$$TS_2' \subseteq TS_2 \tag{7}$$

The key here is to choose the number of samples from TS_1 and TS_2. The more samples selected from TS_1, the stronger the robustness is, but the tampering sensitivity will be reduced; the more samples selected from TS_2, the stronger the algorithm's tampering sensitivity will be. In the extreme case, if the number of samples selected from TS_1 is 0, the method degenerates into the sample construction method in [13]. In the experiment, card (TS_1) = 3000 and card (TS_2) = 10.

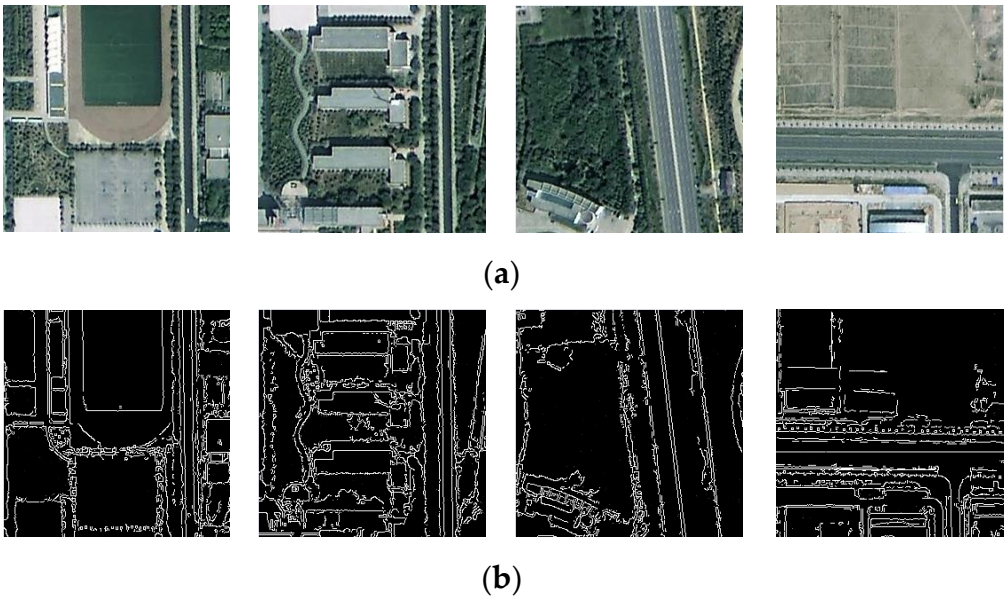


Figure 8. Example of robust edge feature samples made based on the method of [3]: (**a**) Original HRRS images, (**b**) label images for MUM-Net.

4.2. Implementation Details

The proposed MUM-Net model is implemented with the deep learning toolbox Keras 2.2.4 using tensorflow 1.14 as backend in Ubuntu18.04. The computer used was equipped with a single NVIDIA RTX 2080ti GPU with 11 GB GPU memory, 16 GB of RAM and an Intel I7-9700K CPU.

In the training step, we choose batch size = 6 and use the Adam optimizer [51] of which the learning rate is $1e^{-5}$. Due to the preprocessing steps of the algorithm, the size of all images input to MUM-Net is 256×256. MUM-Net contains 12,776,290 parameters, and its size after training is about 146 MB.

4.3. Evaluation Metrics

The goal of our algorithm is to achieve integrity authentication of HRRS images. Therefore, the evaluation metrics of the algorithm are different from other research directions of deep learning such as image segmentation, image classification, and target detection. Based on the analysis of the characteristics of subject-sensitive perceptual hash and the application requirements of HRRS images, it can be concluded that subject-sensitive perceptual hash of HRRS images based on deep learning should meet the following characteristics:

(1) Robustness. Robustness is the most significant difference between perceptual hash and cryptographic hash functions. Robustness means that the perceptual hash sequences of images with the same or similar perceptual content should be the same or similar. Suppose the original HRRS image is represented as I, the HRRS image with unchanged content information is represented as I^1, the perceptual hash function is represented as $H(.)$, and the threshold is T, then:

$$Dis(H(I), H(I^1)) < T \tag{8}$$

(2) Sensitivity to Tampering. Sensitivity to tampering, also known as "collision resistance" or "tampering sensitivity", means that HRRS images with different perceptual content should have different perceptual hash sequences. Suppose the HRRS image with changed content information is represented as I^2, then:

$$Dis(H(I), H(I^2)) \geq T \tag{9}$$

(3) Security. HRRS images are very likely to contain sensitive feature information, so the perceptual hash algorithm must meet security requirements. The security here mainly refers to "one-way": the effective content information of the HRRS image cannot be obtained from the perceptual hash sequence.
(4) High efficiency. High efficiency means that the perceptual hash algorithm can efficiently generate perceptual hash sequences of HRRS images and complete the corresponding authentication.
(5) Tamper localization. For HRRS images with a large amount of data, "tamper localization" should also be paid attention to. This not only enables the user to quickly locate the tampered area, but also reduces the loss caused by the tampering attack, that is, only the tampered area loses its use value, and other areas will not be affected.
(6) Compactness: The perceptual hash sequence should be as compact as possible.

In addition, it should be noted that this paper takes the corrected HRRS image as the research object, so our algorithm should not be robust to the rotation operation.

5. Results and Analysis

In this section, we first give a set of comparative examples to explain the subject-sensitive perceptual hash more vividly. Then, we will analyze the algorithm from the evaluation index of the subject-sensitive hash in Section 4.3.

5.1. Examples of Integrity Authentication

We choose an HRRS image shown in Figure 9a from the WHU building dataset [50] to illustrate the integrity authentication of subject-sensitive perceptual hash. The image is a satellite image of Cairo in Egypt, and its original size is 512 × 512 pixels. We resample it to 256 × 256 to maintain the consistency of the test data. Figure 9a shows the original test image (HRSS image in TIFF format), Figure 9b shows the HRRS image after lossy compression (99% JPEG compression), and Figure 9c shows the HRRS image after adding noise (16 pixels are randomly selected for modification), Figure 9d–f are three examples of tampering that are subject unrelated, and Figure 9g–j are examples of subject-related tampering in which especially Figure 9i,j are tampering that can be easily recognized by the human eye. We not only compare the subject-sensitive perceptual hash algorithm with the existing perceptual hash algorithm, but also compare MUM-Net with other related models.

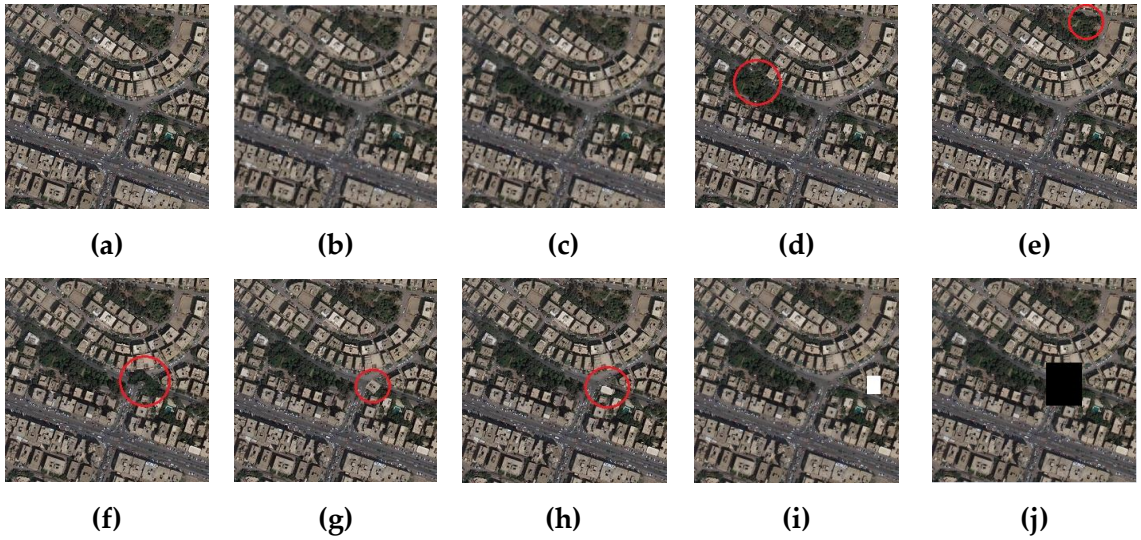


Figure 9. A set of HRRS image integrity authentication examples: (**a**) The original HRRS image (TIFF format); (**b**) lossy compressed image; (**c**) noise added pictures; (**d**–**f**) subject unrelated tampering one-three; (**g**–**j**) subject related tampering one-four.

First, we compare our algorithm with representative image perceptual hash algorithms and perceptual hash algorithms for HRRS images. At present, more representative image perceptual hash algorithms mainly include perceptual hash algorithms based on discrete cosine transform (DCT) [52–54], perceptual hash algorithms based on SIFT feature points [55,56], perceptual hash algorithms based on wavelet transform [57–59] and singular value decomposition (SVD) [60,61]. A brief description of the above comparison algorithms is as follows:

(1) The method based on DCT: the original image is normalized to a size of 64 × 64 pixels, and then DCT transformation is performed to extract low frequency coefficients for quantization. During the authentication process, the threshold is set to 0.02.
(2) The method based on SIFT: First, extract the SFIT feature points of the original image, and then, uniformly select 128 feature points for quantization according to the position information. During the authentication process, the threshold is set to 0.01.
(3) The method based on wavelet transform: the original image is normalized to a size of 64 × 64 pixels, and then DWT transform is performed to extract low frequency coefficients for quantization. During the authentication process, the threshold is set to 0.02.
(4) The method based on SVD: the original image is normalized to a size of 64 × 64 pixels, and then SVD transformation is performed to extract singular values as perceptual features for quantization. During the authentication process, the threshold is set to 0.01.

In the related research for remote sensing images, the algorithm in [14] mainly uses feature points and Zernike moments to extract the features of the HRRS image, which is similar to the existing image perceptual hash, so we will not repeat the comparison. The main theoretical basis of the algorithm in [15] is DWT, which is similar to [57–59] and is no need to repeat the comparison. The method in [49] uses DCT to extract the features of remote sensing images, which is similar to the [52–54]. In [62], there is a perceptual hash algorithm based on a multi-scale strategy, which is highly sensitive to tampering of HRRS images. Therefore, we choose the algorithms in [13] and [62] as comparison algorithms, and the thresholds of both algorithms are set to 0.02 and 0.1 respectively. In addition, it should be pointed out that the algorithm in [13] does not use the training data set of the algorithm in this paper, which is different from the "U-net-based method" in the next comparison.

We can draw preliminary results from Table 1: For operations that do not change the content of HRRS image, such as lossy data compression operations, our algorithm can maintain good robustness; for operations that change the HRRS image content, if the content is changed unrelated to the subject, such as Figure 9d,e, the sensitivity of our algorithm can be appropriately reduced. Of course, as long as the threshold is properly increased, such changes can still be detected; but if the changed content is related to the subject, such as Figure 8g,h, our algorithm has a high sensitivity even at low thresholds. In contrast, although the algorithm in [62] has high tampering sensitivity, it is not robust enough; as the training sample set in this paper is not used, the robustness of the algorithm in [13] needs to be improved, and it cannot distinguish whether the image changes are subject-related.

Next, we compare our MUM-Net with other deep learning models. Specifically, to ensure the fairness of the test, the comparison algorithms use the same process and training data set, but only replaces the corresponding deep learning model. U-net and M-net are the main design reference objects of MUM-Net, and MultiResUNet [36] that is officially published in 2020 is a relatively new and deep network related to U-net, so we select U-net [34], M-net [35], MultiResUNet [36] as a comparison object to verify the effectiveness of MUM-Net. Under the premise of using the same training sample set and the same process, the test results of Figure 9 are shown in Table 2:

Table 1. Comparison with existing perceptual hash algorithms (normalized Hamming distance and integrity authentication result).

Tampering Test	Algorithm Based on DCT	Algorithm Based on SIFT	Algorithm Based on DWT	Algorithm Based on SVD	Algorithm in [13]	Algorithm in [62]	This Algorithm (T = 0.02)	This Algorithm (T = 0.01)
Lossy compressed image	0.045 unpassed	0.02 unpassed	0.017 passed	0.008 passed	0.021 unpassed	0.15 unpassed	0.009 passed	0.009 passed
Noise added image	0.011 passed	0.008 passed	0.015 passed	0.006 passed	0.017 passed	0.14 unpassed	0.015 passed	0.015 unpassed
subject unrelated tampering one	0.000 passed	0.005 passed	0.008 passed	0.002 passed	0.022 unpassed	0.11 unpassed	0.013 passed	0.013 unpassed
subject unrelated tampering two	0.016 passed	0.008 passed	0.014 passed	0.006 passed	0.025 unpassed	0.12 unpassed	0.007 passed	0.007 passed
subject unrelated tampering three	0.033 unpassed	0.012 unpassed	0.019 passed	0.008 passed	0.030 unpassed	0.14 unpassed	0.016 passed	0.016 unpassed
subject related tampering one	0.031 unpassed	0.020 unpassed	0.022 unpassed	0.014 unpassed	0.032 unpassed	0.24 unpassed	0.026 unpassed	0.026 unpassed
subject related tampering two	0.026 unpassed	0.019 unpassed	0.018 passed	0.015 unpassed	0.035 unpassed	0.22 unpassed	0.024 unpassed	0.024 unpassed
subject related tampering three	0.052 unpassed	0.035 unpassed	0.027 unpassed	0.019 unpassed	0.043 unpassed	0.25 unpassed	0.045 unpassed	0.045 unpassed
subject related tampering four	0.068 unpassed	0.044 unpassed	0.051 unpassed	0.025 unpassed	0.048 unpassed	0.27 unpassed	0.051 unpassed	0.051 unpassed

Table 2. Normalized Hamming distances for tamper detection based on different models.

Tampering Test	Algorithm Based on U-Net	Algorithm Based on M-Net	Algorithm Based on MultiResUnet	Algorithm Based on Our Model
Lossy compressed image	0.021	0.014	0.0039	0.011
Noise added pictures	0.015	0.013	0.0026	0.012
subject unrelated tampering one	0.016	0.014	0.014	0.014
subject unrelated tampering two	0.082	0.019	0.005	0.011
subject unrelated tampering three	0.076	0.018	0.009	0.016
subject related tampering one	0.109	0.012	0.008	0.033
subject related tampering two	0.088	0.072	0.027	0.064
subject related tampering three	0.095	0.087	0.078	0.11
subject related tampering four	0.041	0.062	0.037	0.085

It can be seen from Table 2 that for lossy compressed, noise added, and subject-unrelated tampering, the perceptual hash algorithm should have certain robustness, so MultiResUnet performs best; however, for subject related tampering, the perceptual hash algorithm should detect the tampering in the image, so MultiResUnet performed the worst, while U-net performed the best. Our MUM-Net is more balanced in the above two aspects.

We can draw the following preliminary conclusions from Tables 1 and 2:

(1) In the case of using the same training sample data set of this paper and adopting the same algorithm flow, the methods based on deep learning models such as U-net, M-net and MultiResUNet are all "subject-sensitive" to a certain extent. In other words, subject-sensitive perceptual hash can be achieved based on deep learning.

(2) Although the method based on MultiResUNet has good robustness, the tampering sensitivity is insufficient, and malicious tampering cannot be well detected, such as tampering in Figure 9g; the tampering sensitivity of the method based on M-net has increased, but it is still insufficient, if the threshold is set smaller, there is still the possibility of missed detection of malicious tampering, such as the tampering of Figure 9g; U-net-based method has insufficient robustness, and it is too sensitive to subject unrelated changes, such as Figure 9e,f. Although this is not necessarily a bad thing, it means that the "subject-sensitive" of U-net-based method is insufficient; the perceptual hash algorithm based on MUM-Net has a better "subject sensitivity" and can maintain a better balance between algorithm robustness and tampering sensitivity. In short, the perceptual hash algorithm based on MUM-Net has better subject-sensitive characteristics.

This section describes the subject-sensitive perceptual hash through a set of examples, and shows that MUM-Net can better achieve subject-sensitive perceptual hash through comparative experiments. Next, we will analyze our algorithm more comprehensively.

5.2. Performance of Perceptual Robustness

The robust testing of subject-sensitive perceptual hash requires a relatively large amount of test data. We build our test data set based on the data sets of TS_1 generated in Section 4, AID [63], DOTA [64], UC Merced Land-Use Dataset [65], SIRI-WHU Dataset [66], WHU-RS19 Dataset [67] and the test data of [13] and [62]. The following steps are used to construct a data set for testing robustness:

First, we select part of the data that is not used as the training sample in the data set TS_1 as the testing data to test the robustness of the algorithm, denote as ${TS_1}^{\mathrm{T}}$. The number of images we choose is 200, which means:

$$\mathrm{card}(TS_1^{\mathrm{T}}) = 200 \tag{10}$$

Second, from the four datasets of AID (each image size is 600 × 600 pixels), UC Merced Land-Use (each image size is 256 × 256 pixels), and WHU-RS19 (each image size is 600 × 600 pixels), we respectively select 200 HRRS images and record them as TS_{AID}, TS_{Merced}, TS_{RS19}, and each image was resized to 256 × 256 pixels and saved in TIFF format. Obviously:

$$\mathrm{card}(TS_{\mathrm{AID}}) = \mathrm{card}(TS_{\mathrm{Merced}}) = \mathrm{card}(TS_{\mathrm{RS19}}) = 200 \tag{11}$$

Examples of the testing image in TS_{AID}, TS_{Merced} and TS_{RS19} are shown in Figure 10.

Thirdly, we select 40 images from the DOTA dataset, and divide each image into grid cells. The size of the grid cell after division is adjusted to 256 × 256 pixels and saved as TIFF format. Compared with the images of AID and WHU-RS19, the images in the DOTA dataset are relatively large, so the grid division process needs to be used in the preprocessing stage and each grid cell can be used as a test image. In addition, we eliminated some grid cells that did not meet the requirements. For example, some grid cells only contained lakes. In this way, the test data set containing 411 images with the size of 256 × 256 pixels is finally obtained and is saved as TIFF format, which is recorded as TS_{DOTA}. Similarly, we obtained 962 images with the size of 256 × 256 pixels from the test data of [13] and [62] (excluding the grid cells in which grassland, mountains, or water occupy most of the area), which is denoted as TS_3^{T}.

Examples of the testing image in TS_{DOTA} and TS_3 are shown in Figure 11.

Format conversion and digital watermark embedding are two common operations that do not change image content and are often used to test the robustness of perceptual hash algorithms. Similar to the previous related research [13–15,62], Experiments show that four deep learning-based algorithms, namely, U-net-based algorithm, M-net-based algorithm, MultiResUnet-based algorithm, and MUM-Net-based algorithm, can maintain 100% robustness to TIFF to BMP format conversion.

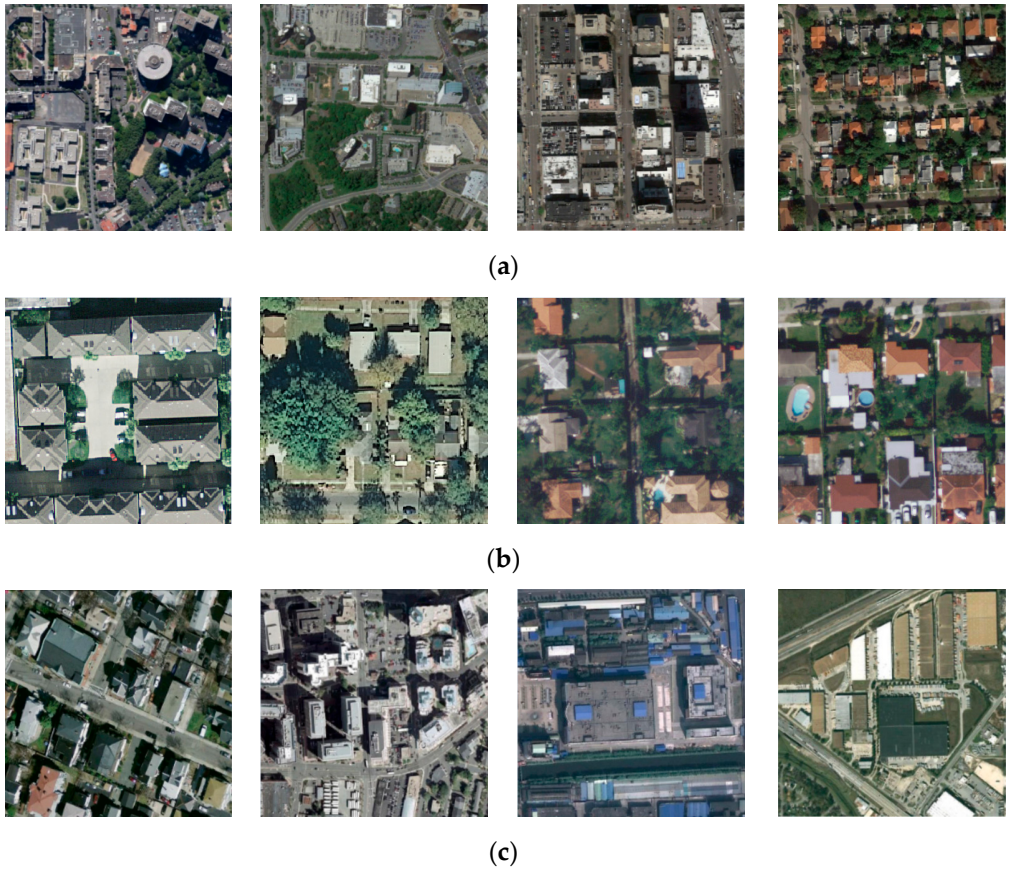


Figure 10. Examples of the testing image: (**a**) TS_{AID}, (**b**) TS_{Merced}, (**c**) TS_{RS19}.

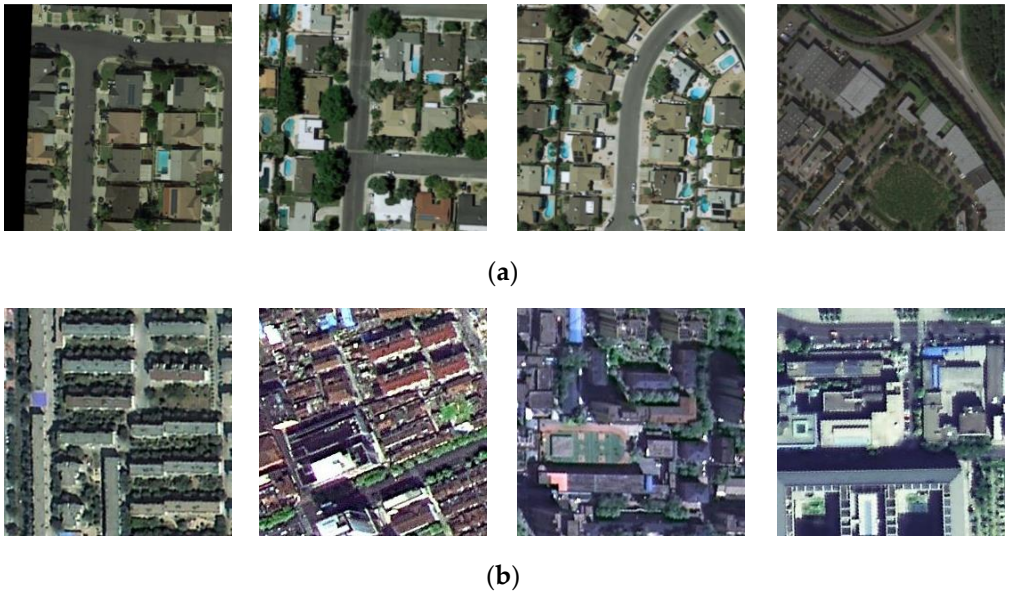


Figure 11. Examples of the testing image: (**a**) TS_{DOTA}, (**b**) TS_3^{T}.

For the digital watermarking algorithm, the least significant bit (LSB) algorithm also changes the HRRS image very slightly, which is similar to the format conversion between TIFF and BMP. Therefore, we choose to embed the watermark in the next lowest bit of the HRRS image pixel, and then test the robustness of the algorithm. The results are shown in Table 3.

Table 3. Comparison of robustness test of digital watermarking embedding (T_h is set to 0.01).

Manipulation	Algorithm Based on U-Net	Algorithm Based on M-Net	Algorithm Based on MultiResUnet	Algorithm Based on Our Model
${TS_1}^{\mathrm{T}}$	100%	100%	100%	100%
TS_{AID}	99.5%	100%	100%	100%
TS_{Merced}	98.0%	100%	100%	100%
TS_{RS19}	99.5%	99.5%	100%	99.5%
TS_{DOTA}	98.8%	100%	100%	100%
${TS_3}^{\mathrm{T}}$	95.8%	99.6%	100%	99.8%

In this paper, we use the proportion of grid cells whose normalized hamming distance is lower than the threshold to describe the experimental results. For example, as the threshold is set to 0.01 in Table 3, the perceptual hash algorithm based on U-net maintains robustness to 99.5% for TS_{AID}, that is, 99.5% of the images are not detected.

It can be seen from Table 3 that the algorithm in this paper has good robustness for digital watermark embedding, and the digital watermarking algorithm for testing is not the least significant bit algorithm with the least changes to data.

Next, we test the robustness of the algorithm with two operations: noise embedding and data compression. Noise embedding will modify the pixels of the image. If there is too much noise embedded, the content of the image will change too much, so we take pepper and salt noise embedding (16 pixels are randomly selected for modification) as an example. For data compression, we take JPEG compression as an example. The results are shown in Tables 4 and 5.

Table 4. Comparison of robustness test of JPEG compression (T_h is set to 0.02).

Manipulation	Algorithm Based on U-Net	Algorithm Based on M-Net	Algorithm Based on MultiResUnet	Algorithm Based on Our Model
${TS_1}^{\mathrm{T}}$	77.0%	86.5%	92.0%	89.5%
TS_{AID}	81.0%	88.0%	94.0%	91.0%
TS_{Merced}	75.5%	84.0%	91.5%	89.5%
TS_{RS19}	78.5%	87.0%	93.0%	90.5%
TS_{DOTA}	82.5%	84.5%	95.6%	88.5%
${TS_3}^{\mathrm{T}}$	75.1%	86.3%	91.6%	90.3%

Table 5. Comparison of robustness test of adding noise (T_h is set to 0.02).

Manipulation	Algorithm Based on U-Net	Algorithm Based on M-Net	Algorithm Based on MultiResUnet	Algorithm Based on Our Model
${TS_1}^{\mathrm{T}}$	81.5%	89.0%	94.0%	93.0%
TS_{AID}	84.0%	90.5%	95.0%	92.0%
TS_{Merced}	82.5%	90.0%	94.5%	92.0%
TS_{RS19}	82.5%	89.0%	95.5%	91.5%
TS_{DOTA}	83.0%	87.6%	95.1%	90.5%
${TS_3}^{\mathrm{T}}$	79.1%	90.2%	94.0%	91.9%

It can be seen from Tables 4 and 5 that the MUM-Net-based algorithm in this paper is more robust than U-net-based algorithm and M-net-based algorithm. The perceptual hash algorithm does not simply emphasize the robustness, but should have better robustness and tampering sensitivity. Therefore, although Tables 4 and 5 show that the algorithm based on MultiResUnet is more robust than MUM-Net, the algorithm based on MultiResUnet has a large deficiency in tamper sensitivity according to the experimental results of tamper sensitivity in Section 5.3, that is to say, the overall performance of the algorithm in this paper is better.

In fact, robustness itself is a relative concept: different threshold settings often result in different robustness test results. Taking the noise embedding operation as an example, if we increase the threshold from 0.02 to 0.05, the comparison results of the robustnes are shown in Table 6:

Table 6. Comparison of robustness test of adding noise (T_h is set to 0.05, different with Table 5).

Manipulation	Algorithm Based on U-Net	Algorithm Based on M-Net	Algorithm Based on MultiResUnet	Algorithm Based on Our Model
${TS_1}^{\mathrm{T}}$	93.0%	94.0%	99.0%	97.5%
TS_{AID}	92.5%	94.5%	99.5%	95.5%
TS_{Merced}	91.5%	95.5%	99.0%	95.5%
TS_{RS19}	90.5%	94.0%	98.5%	95.0%
TS_{DOTA}	88.6%	91.2%	96.4%	94.4%
${TS_3}^{\mathrm{T}}$	89.5%	92.9%	95.5%	94.8%

It can be seen from Tables 5 and 6 that the robustness of the perceptual hash algorithm is related to the strength requirements of the algorithm in the authentication phase. In short, compared to the U-net-based perceptual hash algorithm [13], our algorithm is more robust to noise addition and JPEG compression by about 10%.

In practical applications, if the robustness requirement is high, a relatively large threshold can be set; conversely, if the robustness requirement is low and the tampering sensitivity requirement is higher, the relatively small threshold should be set.

5.3. Performance of Sensitivity to Tampering

Although robustness is the greatest advantage and characteristic of perceptual hash over cryptographic hash functions, perceptual hash needs to be able to identify whether data has been tampered with like cryptographic hash functions, that is, perceptual hash should also have good sensitivity to tampering.

The algorithm in this paper is not completely equivalent to the conventional perceptual hash algorithm, and its "subject-sensitive" feature should also be reflected in terms of sensitivity to tampering. Therefore, the tamper sensitivity test needs to be divided into two types of tampering: "subject-related" and "subject-unrelated".

Firstly, we specially made a selection of images from the dataset of TS_3^{T} for "subject-unrelated" modification, and the modified content is "trees", "lawn", etc., that is, changes that are subject unrelated. The testing dataset composed of the modified images is recorded as ${TS_3}^{\mathrm{SU}}$, and card(${TS_3}^{\mathrm{SU}}$) = 200. Examples of the testing image in ${TS_3}^{\mathrm{SU}}$ are shown in Figure 12.

Since the existing perceptual hash algorithm based on conventional feature extraction cannot achieve subject-sensitive perceptual hash, we compare our proposed algorithm with the algorithms based on U-net, M-net and MultiResUnet. The results are shown in Table 7. Here, we describe the tampering sensitivity in terms of omission factor under different threshold. For example, when the threshold is set to 0.01, the perceptual hash algorithm based on MUM-Net maintains robustness to 27.5% of the images, that is, 27.5% of image changes (tampering) are not detected.

We can draw the conclusions from Table 7: For the subject unrelated tampering, the tampering sensitivity of our algorithm is relatively reduced, and it also can be understood that the robustness has been enhanced. This is consistent with the concept of subject-sensitive perceptual hash. Of course, even if tampering is not related to the subject, as long as a lower threshold is set, our algorithm can still be effectively detected.

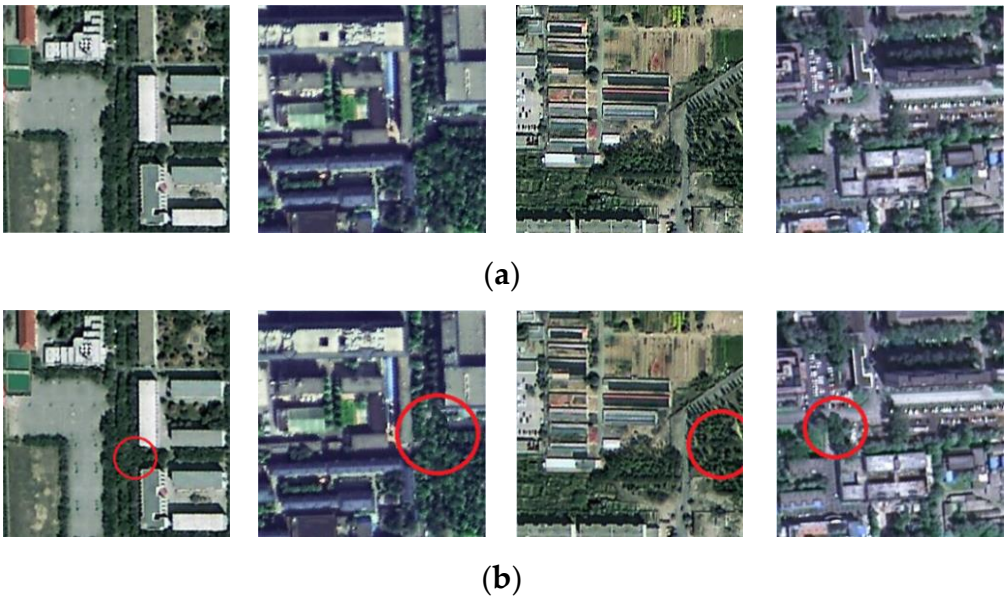


Figure 12. Examples of the subject unrelated tampering: (**a**) Original HRRS images, (**b**) subject unrelated tampering.

Table 7. Omission factor of sensitivity to tampering with subject unrelated changing.

Threshold	Algorithm Based On U-Net	Algorithm Based on M-Net	Algorithm Based on MultiResUnet	Algorithm Based on Our Model
$T = 0.01$	22.5%	34.5%	96%	27.5%
$T = 0.02$	39.0%	45.5%	97.5%	40.5%
$T = 0.03$	53.5%	58.0%	100%	51.5%
$T = 0.05$	84.0%	87.0%	100%	76.5%

Secondly, we made a selection of images from the dataset of ${TS_1}^{\mathrm{T}}$ and ${TS_3}^{\mathrm{T}}$ for "subject-related" modification. The modification content is subject related and mainly includes three types of modifications: adding object, deleting object and modifying object. Each type of modification contains 200 instances. The testing dataset composed of the modified images is recorded as ${TS_3}^{\mathrm{SR}}$ and $\mathrm{card}({TS_3}^{\mathrm{SR}}) = 600$. Examples of the testing image in ${TS_1}^{\mathrm{SU}}$ are shown in Figure 13.

For the above types of subject related tampering, we also use the omission factor under different threshold to describe the sensitivity to tampering and compare our algorithm with the algorithms based on U-net, M-net and MultiResUnet. The results are shown in Tables 8–10.

Table 8. Omission factor of sensitivity to tampering with changing object.

Threshold	Algorithm Based on U-Net	Algorithm Based on M-Net	Algorithm Based on MultiResUnet	Algorithm Based on Our Model
$T = 0.01$	0.0%	1.0%	4.5%	0.0%
$T = 0.02$	6.5%	16.5%	24.0%	15.0%
$T = 0.03$	22.0%	28.5%	38.0%	26.5%
$T = 0.04$	31.0%	37.0%	52.5%	33.0%

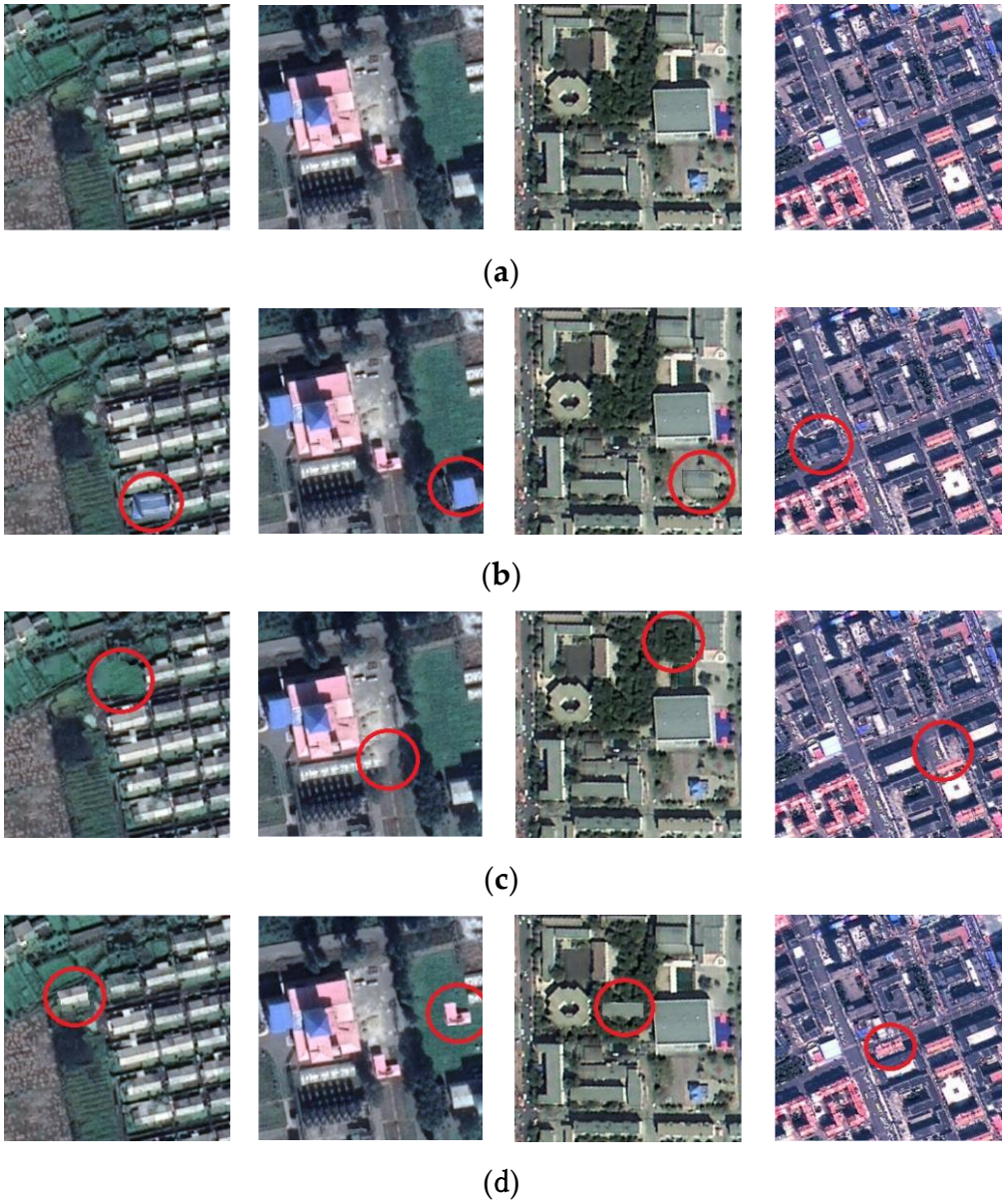


Figure 13. Example of subject related tampering: (**a**) Original HRRS images, (**b**) changing object, (**c**) removing object, (**d**) appending object.

Table 9. Omission factor of sensitivity to tampering with removing object.

Threshold	Algorithm Based on U-Net	Algorithm Based on M-Net	Algorithm Based on MultiResUnet	Algorithm Based On Our Model
$T = 0.01$	0.0%	5.0%	11.0%	0.5%
$T = 0.02$	2.5%	7.0%	15.0%	3.5%
$T = 0.03$	6.5%	12.5%	25.5%	8.0%
$T = 0.04$	13.5%	20.5%	39.0%	15.0%

Table 10. Omission factor of sensitivity to tampering with appending object.

Threshold	Algorithm Based on U-Net	Algorithm Based on M-Net	Algorithm Based on MultiResUnet	Algorithm Based on Our Model
$T = 0.01$	0.0%	1.5%	23.0%	1.0%
$T = 0.02$	9.0%	8.5%	38.5%	5.5%
$T = 0.03$	11.0%	10.0%	44.5%	9.0%
$T = 0.04$	12.5%	14.0%	56.0%	11.5%

It can be clearly seen from Tables 8–10 that the lower the threshold is set, the stronger the tampering sensitivity of each algorithm. What's more, the sensitivity to subject-related tampering of our algorithm gradually approaches U-net as the threshold increases.

Further analysis of the results in Tables 8–10 shows that the tampering sensitivity of our algorithm is generally only slightly weaker than the U-net-based algorithm, but stronger than the M-net-based and MultiResUnet-based algorithms., but in the experiment in Section 5.2, the robustness of the U-net-based algorithm is the worst of the four algorithms. The tampering sensitivity of the algorithm based on MultiResUnet is relatively weak, while in the experiment in Section 5.2, the robustness of the algorithm based on MultiResUnet is the best. Therefore, our algorithm achieves better overall performance in the contradictory attributes of tampering sensitivity and robustness.

In summary, we can draw the following conclusions: For the subject unrelated tampering, the tampering sensitivity of our algorithm is relatively reduced, and it can be understood that the robustness has been enhanced; but for the subject related tampering, our algorithm shows a stronger tampering sensitivity.

5.4. Analysis of Algorithm Security

The security of the algorithm mainly relies on the following two aspects:

(1) Due to the complex multi-layer nonlinear network structure, it is difficult to interpret convolutional neural networks from visual semantics [68,69]. Although the interpretable difficulty of a deep learning model is a disadvantage in other deep learning fields, the difficulty of interpretation can well guarantee the security of the algorithm, that is, even in the case of obtaining a hash sequence, it is difficult to obtain the input HRRS image in reverse.

(2) The algorithm uses the AES algorithm to encrypt the perceptual features in the compression coding stage, and the security of AES has long been widely recognized. Therefore, the application of the AES algorithm further strengthens the security of the algorithm.

5.5. Tampering Location Analysis

The process of tampering location is as follows: First, after the authenticating party receives the image to be authenticated and the perceptual hash sequence (denoted as PH) of the original image, it calculates the perceptual hash sequence of the image to be authenticated according to the algorithm flow in Section 3.1, that is, the image is divided into grids to generate hash sequence of each grid cell (denoted as PH'_{ij}), and the perceptual hash sequence (denoted as PH') of the image. Next, if PH and PH' are exactly equal, then image has passed the integrity authentication and the authentication process ends; otherwise, the normalized Hamming distance between PH_{ij} and PH'_{ij} is calculated one by one. If the normalized Hamming distance is greater than the threshold, then the content of the grid cell has been significantly tampered. After verifying the integrity of all grid cells one by one, the corresponding tampering location results are obtained.

The granularity of the algorithm's tamper localization depends on the granularity of grid division. Obviously, the finer the granularity of the tampering of the algorithm, the greater the amount of data

in the perceived hash sequence, which means that the compactness of the algorithm will be greatly affected. Therefore, the ability to tamper positioning is not as strong as possible.

Figure 14 shows an example of tamper localization of our algorithm. Figure 14a shows original HRRS images, Figure 14b shows the result of tampering location, Figure 14c shows original grid cell, and Figure 14d shows the corresponding tampered grid cell.

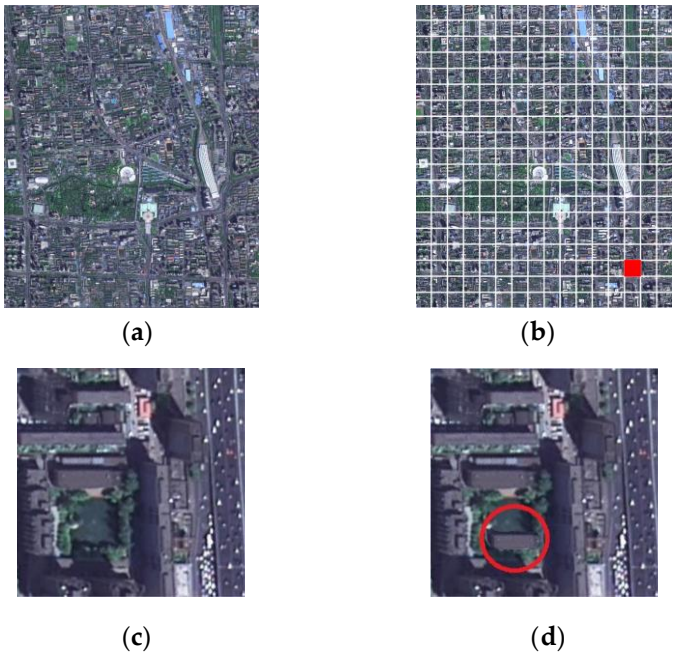


Figure 14. Example of tamper location: (**a**) Original HRRS image, (**b**) Result of Tamper location, (**c**) Original grid cell, (**d**) Tampered grid cell.

6. Discussion

In this research, we introduce the concept of "subject-sensitive perceptual hash" based on the analysis of the problems in integrity authentication of HRRS image. Subject-sensitive perceptual hash, which is a special case of conventional perceptual hash, aims to achieve subject-biased integrity authentication. However, it is very difficult to implement subject-sensitive perceptual hash based on traditional image processing methods, because most traditional image feature extraction methods cannot achieve "subject-biased" feature extraction.

To achieve subject-sensitive perceptual hash, we propose a novel deep learning model named MUM-Net to extract the robust features of HRRS images. MUM-Net extracts as much feature information as possible through multi-scale input. At the same time, it uses M-net's process of denoising fingerprint images to eliminate unnecessary features and enhance the robustness of the algorithm. Compared with the original U-net, the robustness of MUM-Net has been greatly improved. Compared with M-net, MUM-Net is more sensitive to tampering.

In the experiments of this paper, we take building information as an example to construct the training data set and conduct an experimental analysis to illustrate the effectiveness of the "subject sensitive perceptual hash". Combining the evaluation metrics of subject-sensitive perceptual hash in Section 4.3, we analyze the experimental results of Section 5 and can draw the following conclusions:

(1) Robustness. The sample learning ability of MultiResUnet is the strongest among all models. Most of the edge features it extracts are the edges of buildings, so the perceptual hash algorithm based on MultiResUnet is the most robust. MUM-Net is more robust than M-net and U-net, while M-net is better than U-net. From Tables 3–6, we can get the following robustness ranking:

MultiResUnet > MUM-Net > M-net >U-net

(2) Tampering sensitivity. Although robustness is the biggest advantage of perceptual hash over cryptographic hash, if the sensitivity of tampering is insufficient, it means that perceptual hash cannot detect possible malicious tampering, and it will not be able to meet the requirements of integrity authentication. Although most of the edges detected by MultiResUnet are the edges of buildings, many relevant edges are missed, which is very detrimental to its tampering sensitivity. From Tables 7–10, we can draw the conclusion that MultiResUnet's tampering sensitivity is the worst among all models, and M-net also has similar problems. The sensitivity of U-net is the best, even stronger than MUM-Net. Therefore, we can rank the tampering sensitivity as follows:

$$\text{U-net} > \text{MUM-Net} > \text{M-net} > \text{MultiResUnet}$$

(3) Security and tampering positioning. Since the algorithm processes adopted by each deep learning model are similar, but differ in the perceptual feature extraction stage, there is not much difference in the security and tampering positioning of MUM-Net, U-net, M-net and MultiResUnet.

In summary, we can draw this conclusion: our algorithm not only achieves subject-sensitive perceptual hash, but also achieves a better overall performance between tampering sensitivity and robustness, which is better than the algorithms based on U-net, M-net and MultiResUnet.

In addition, since the perceptual hash algorithm based on traditional feature extraction methods does not have the ability to learn samples, that is, it cannot achieve "subject-biased" integrity authentication, so only the perceptual hash algorithms based on different deep learning models is discussed here. Moreover, in [13], it has made a detailed comparison and analysis of the U-net-based perceptual hash and the traditional perceptual hash. The results in [13] show that the comprehensive performance of the U-net-based perceptual hash is stronger than traditional perceptual hash algorithm.

7. Conclusions

In this paper, we introduce the concept of "subject-sensitive perceptual hash" which is a special case of conventional perceptual hash, and we developed a novel deep learning model named MUM-Net to achieve subject-sensitive perceptual hash for HRRS image authentication. This network can effectively extract subject-sensitive features, which can effectively detect malicious tampering and maintain good robustness. What is more, we propose a construction method of training sample set, which effectively use existing data sets. In the experiments, we not only compare our algorithm with the traditional perceptual hash algorithm, but also compare it with the perceptual hash algorithms based on U-net, M-net and MultiResUnet. The results show that our algorithm is more robust by about 10% compared to existing U-net-based perceptual hash algorithms. What's more, our algorithm not only achieves subject-sensitive perceptual hash, but also achieves a better overall performance between tampering sensitivity and robustness.

In future work, we intend to study the applicability of the perceptual hash algorithm to different resolution images, and the method to locate the tampered area intelligently based on deep learning.

Author Contributions: Kaimeng Ding conceived the idea and worked together with Qin Xu to design the scheme; Yueming Liu assisted with the study design and the experiments; Fuqiang Lu participated in the collection and collation of experimental data. All authors reviewed the manuscript. All authors have read and agreed to the published version of the manuscript.

Funding: This study is patricianly supported by the grants from: (a) the National Natural Science Foundation of China (Grant Nos. 41801303, 41901323); (b) the Jiangsu Province Science and Technology Support Program (Grant No. BK20170116); (c) the Scientific Research Hatch Fund of Jinling Institute of Technology (Grant Nos. jit-fhxm-201604, jit-b-201520; jit-b-201645); and (d) the Qing Lan Project.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Hu, F.; Xia, G.S.; Hu, J.; Zhang, L. Transferring Deep Convolutional Neural Networks for the Scene Classification of High-Resolution Remote Sensing Imagery. *Remote Sens.* **2015**, *7*, 14680–14707. [CrossRef]
2. Yang, G.; Zhang, Q.; Zhang, G. EANet: Edge-Aware Network for the Extraction of Buildings from Aerial Images. *Remote Sens.* **2020**, *12*, 2161. [CrossRef]
3. Zhang, H.; Yang, W.; Yu, H.; Zhang, H.; Xia, G.S. Detecting Power Lines in UAV Images with Convolutional Features and Structured Constraints. *Remote Sens.* **2019**, *11*, 1342. [CrossRef]
4. Hoeser, T.; Kuenzer, C. Object Detection and Image Segmentation with Deep Learning on Earth Observation Data: A Review-Part I: Evolution and Recent Trends. *Remote Sens.* **2020**, *12*, 1667. [CrossRef]
5. Wang, Y.D.; Li, Z.W.; Zeng, C.; Xia, G.S.; Shen, H.F. An Urban Water Extraction Method Combining Deep Learning and Google Earth Engine. *IEEE J. Select. Top. Appl. Earth Observ. Remote Sens.* **2020**, *13*, 768–781. [CrossRef]
6. Tavakkoli Piralilou, S.; Shahabi, H.; Jarihani, B.; Ghorbanzadeh, O.; Blaschke, T.; Gholamnia, K.; Meena, S.R.; Aryal, J. Landslide Detection Using Multi-Scale Image Segmentation and Different Machine Learning Models in the Higher Himalayas. *Remote Sens.* **2019**, *11*, 2575. [CrossRef]
7. Xu, L.; Chen, Y.Y.; Pan, J.W.; Gap, A. Multi-Structure Joint Decision-Making Approach for Land Use Classification of High-Resolution Remote Sensing Images Based on CNNs. *IEEE Access.* **2020**, *8*, 42848–42863. [CrossRef]
8. Xu, S.H.; Mu, X.D.; Ke, B.; Wang, X.R. Dynamic Monitoring of Military Position based on Remote Sensing Image. *Remote Sensing Technol. Appl.* **2014**, *29*, 511–516.
9. Zhang, C.; Wei, S.; Ji, S.; Lu, M. Detecting Large-Scale Urban Land Cover Changes from Very High Resolution Remote Sensing Images Using CNN-Based Classification. *ISPRS Int. J. Geo-Inf.* **2019**, *8*, 189. [CrossRef]
10. Li, J.; Pei, Y.; Zhao, S.; Xiao, R.; Sang, X.; Zhang, C. A Review of Remote Sensing for Environmental Monitoring in China. *Remote Sens.* **2020**, *12*, 1130. [CrossRef]
11. Niu, X.M.; Jiao, Y.H. An Overview of Perceptual Hashing. *Acta Electron. Sin.* **2008**, *36*, 1405–1411.
12. Qin, C.; Sun, M.; Chang, C.C. Perceptual hashing for color images based on hybrid extraction of structural features. *Signal. Process.* **2018**, *142*, 194–205. [CrossRef]
13. Ding, K.M.; Yang, Z.D.; Wang, Y.Y.; Liu, Y.M. An improved perceptual hash algorithm based on u-net for the authentication of high-resolution remote sensing image. *Appl. Sci.* **2019**, *9*, 2972. [CrossRef]
14. Zhang, X.G.; Yan, H.W.; Zhang, L.M.; Wang, H. High-Resolution Remote Sensing Image Integrity Authentication Method Considering Both Global and Local Features. *ISPRS Int. J. Geo-Inf.* **2020**, *9*, 254. [CrossRef]
15. Ding, K.M.; Zhu, Y.T.; Zhu, C.Q.; Su, S.B. A perceptual Hash Algorithm Based on Gabor Filter Bank and DWT for Remote Sensing Image Authentication. *J. China Railw. Soc.* **2016**, *38*, 70–76.
16. Du, L.; Ho, A.T.S.; Cong, R. Perceptual hashing for image authentication: A survey. *Sig. Process. Image Commun.* **2020**, *81*, 115713. [CrossRef]
17. Tang, Z.J.; Huang, Z.Q.; Zhang, X.Q.; Lao, H. Robust image hashing with multidimensional scaling. *Sig. Process.* **2017**, *137*, 240–250. [CrossRef]
18. Yan, C.P.; Pun, C.M.; Yuan, X.C. Quaternion-based image hashing for adaptive tampering localization. *IEEE Trans. Inform. Forens. Secur.* **2016**, *11*, 2664–2677. [CrossRef]
19. Lv, X.; Wang, Z.J. Perceptual image hashing based on shape contexts and local feature points. *IEEE Trans. Inform. Forens. Secur.* **2012**, *7*, 1081–1093. [CrossRef]
20. Liu, Z.Q.; Li, Q.; Liu, J.R.; Peng, X.Y. SIFT based image hashing algorithm. *Chin. J. Sci. Instrum.* **2011**, *32*, 2024–2028.
21. Monga, V.; Evans, B.L. Perceptual image hashing via feature points: Performance evaluation and tradeoffs. *Ieee Trans. Image Process.* **2006**, *15*, 3452–3465. [CrossRef] [PubMed]
22. Khelifi, F.; Jiang, J. Analysis of the security of perceptual image hashing based on non-negative matrix factorization. *Ieee Sig. Process. Lett.* **2009**, *17*, 43–46. [CrossRef]
23. Liu, H.; Xiao, D.; Xiao, Y.P.; Zhang, Y.S. Robust image hashing with tampering recovery capability via low-rank and sparse representation. *Multimed. Tools Appl.* **2016**, *75*, 7681–7696. [CrossRef]
24. Sun, R.; Zeng, W. Secure and robust image hashing via compressive sensing. *Multimed. Tools Appl.* **2014**, *70*, 1651–1665. [CrossRef]

25. Tang, Z.J.; Zhang, X.Q.; Huang, L.Y.; Dai, Y.M. Robust image hashing using ring-based entropies. *Sig. Process.* **2013**, *93*, 2061–2069. [CrossRef]
26. Chen, Y.; Yu, W.; Feng, J. Robust image hashing using invariants of Tchebichef moments. *Optik* **2014**, *125*, 5582–5587. [CrossRef]
27. Sajjad, M.; Haq, I.U.; Lloret, J.; Ding, W.P.; Muhammad, K. Robust image hashing based efficient authentication for smart industrial environment. *Ieee Trans. Industr. Informat.* **2019**, *15*, 6541–6550. [CrossRef]
28. Rostami, M.; Kolouri, S.; Eaton, E.; Kim, K. Deep Transfer Learning for Few-Shot SAR Image Classification. *Remote Sens.* **2019**, *11*, 1374. [CrossRef]
29. Fang, B.; Kou, R.; Pan, L.; Chen, P. Category-Sensitive Domain Adaptation for Land Cover Mapping in Aerial Scenes. *Remote Sens.* **2019**, *11*, 2631. [CrossRef]
30. Pires de Lima, R.; Marfurt, K. Convolutional Neural Network for Remote-Sensing Scene Classification: Transfer Learning Analysis. *Remote Sens.* **2020**, *12*, 86. [CrossRef]
31. Weinstein, B.G.; Marconi, S.; Bohlman, S.; Zare, A.; White, E. Individual Tree-Crown Detection in RGB Imagery Using Semi-Supervised Deep Learning Neural Networks. *Remote Sens.* **2019**, *11*, 1309. [CrossRef]
32. Ghorbanzadeh, O.; Blaschke, T.; Gholamnia, K.; Meena, S.R.; Tiede, D.; Aryal, J. Evaluation of Different Machine Learning Methods and Deep-Learning Convolutional Neural Networks for Landslide Detection. *Remote Sens.* **2019**, *11*, 196. [CrossRef]
33. Jiang, C.; Pang, Y. Perceptual image hashing based on a deep convolution neural network for content authentication. *J. Electron. Imag.* **2018**, *27*, 1–11. [CrossRef]
34. Ronneberger, O.; Fischer, P.; Brox, T. U-net: Convolutional networks for biomedical image segmentation. In Proceedings of the 18th International Conference on Medical Image Computing and Computer-Assisted Intervention, Munich, Germany, 5–9 October 2015; pp. 234–241.
35. Adiga, V.; Sivaswamy, J. FPD-M-net: Fingerprint Image Denoising and Inpainting Using M-Net Based Convolutional Neural Networks. *Arxiv Comp. Vis. Pattern Recog.* **2019**, 51–61.
36. Ibtehaz, N.; Rahman M, S. MultiResUNet: Rethinking the U-Net architecture for multimodal biomedical image segmentation. *Neural Net.* **2020**, *121*, 74–87. [CrossRef] [PubMed]
37. Long, J.; Shelhamer, E.; Darrell, T. Fully convolutional networks for semantic segmentation. In Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, USA, 7–12 June 2015; pp. 3431–3440.
38. Bengio, Y.; LeCun, Y. Scaling learning algorithms towards AI. *Large-Scale Kern. Mach.* **2007**, *34*, 1–41.
39. Xia, X.; Kulis, B. W-net: A deep model for fully unsupervised image segmentation. *arXiv* **2017**, arXiv:1711.08506.
40. Li, X.L.; Wang, Y.Y.; Tang, Q.S.; Fan, Z.; Yu, J.H. Dual U-Net for the Segmentation of Overlapping Glioma Nuclei. *Ieee Access* **2019**, *7*, 84040–84052. [CrossRef]
41. Francia, G.A.; Pedraza, C.; Aceves, M.; Tovar-Arriaga, S. Chaining a U-Net with a Residual U-Net for Retinal Blood Vessels Segmentation. *IEEE Access.* **2020**, *8*, 38493–38500. [CrossRef]
42. Zhao, H.; Qi, X.; Shen, X.; Shi, J.; Jia, J. ICNet for Real-Time Semantic Segmentation on High-Resolution Images. In Proceedings of the 15th European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 3735–3739.
43. Zhang, J.W.; Jin, Y.Z.; Xu, J.L.; Xu, X.W.; Zhang, Y.C. MDU-Net: Multi-scale Densely Connected U-Net for biomedical image segmentation. *arXiv* **2018**, arXiv:1812.00352.
44. Ren, W.Q.; Pan, J.S.; Zhang, H.; Cao, X.C.; Yang, M.H. Single image dehazing via multi-scale convolutional neural networks with holistic edges. *Int. J. Comp. Vis.* **2020**, *128*, 240–259. [CrossRef]
45. Villamizar, M.; Canévet, O.; Odobez, J.M. Multi-scale sequential network for semantic text segmentation and localization. *Recognit. Lett.* **2020**, *129*, 63–69. [CrossRef]
46. Lin, T.; Goyal, P.; Girshick, R.; He, K.; Dollar, P. Focal Loss for Dense Object Detection. In Proceedings of the IEEE International Conference on Computer Vision (ICCV), Venice, Italy, 22–29 October 2017; pp. 2999–3007.
47. Ji, B.; Ren, J.J.; Zheng, X.J.; Tan, C.; Ji, R.; Zhao, Y.; Liu, K. A multi-scale recurrent fully convolution neural network for laryngeal leukoplakia segmentation. *Process. Control.* **2020**, *59*, 101913. [CrossRef]
48. Lin, T.Y.; Goyal, P.; Girshick, R.; He, K.M.; Dollar, P. Focal Loss for Dense Object Detection. *IEEE Trans. Pattern Anal. Mach. Intell.* **2020**, *42*, 318–327. [CrossRef]
49. Ding, K.M.; Zhu, C.Q.; Lu, F.Q. An adaptive grid partition based perceptual hash algorithm for remote sensing image authentication. *Wuhan Daxue Xuebao* **2015**, *40*, 716–720.

50. Ji, S.P.; Wei, S.Y. Building extraction via convolutional neural networks from an open remote sensing building dataset. *Acta Geod. Et Cartogr. Sinica.* **2019**, *48*, 448–459.
51. Kingma, D.; Ba, J. Adam: A Method for Stochastic Optimization. In Proceedings of the 3rd International Conference for Learning Representations (ICLR), San Diego, CA, USA, 7–9 May 2015.
52. Zhang, Y.D.; Tang, S.; Li, J.T. Secure and Incidental Distortion Tolerant Digital Signature for Image Authentication. *J. Comput. Sci. Technol.* **2007**, *22*, 618–625. [CrossRef]
53. Fang, W.; HU, H.M.; Hu, Z.; Liao, S.C.; Li, B. Perceptual hash-based feature description for person re-identification. *Neurocomputing* **2018**, *272*, 520–531. [CrossRef]
54. Wang, H.; Wang, H.X. Perceptual Hashing-Based Image Copy-Move Forgery Detection. *Secur. Commun. Netw.* **2018**, *2018*, 1–11. [CrossRef]
55. Singh, K.M.; Neelima, A.; Tuithung, T.; Singh, K.M. Robust Perceptual Image Hashing using SIFT and SVD. *curr. Sci.* **2019**, *8*, 117.
56. Ouyang, J.L.; Liu, Y.Z.; Shu, H.Z. Robust Hashing for Image Authentication Using SIFT Feature and Quaternion Zernike Moments. *Multimed. Tool Appl.* **2017**, *76*, 2609–2626. [CrossRef]
57. Lu, C.S.; Liao, H.Y.M. Structural digital signature for image authentication: An incidental distortion resistant scheme. *Ieee Trans. Multimed.* **2003**, *5*, 161–173.
58. Zhang, Q.H.; Xing, P.F.; Huang, Y.B.; Dong, R.H.; Yang, Z.P. An efficient speech perceptual hashing authentication algorithm based on DWT and symmetric ternary string. *Int. J. Informat. Comm. Technol.* **2018**, *12*, 31–50.
59. Yang, Y.; Zhou, J.; Duan, F.; Liu, F.; Cheng, L.M. Wave atom transform based image hashing using distributed source coding. *J. Inf. Secur. Appl.* **2016**, *31*, 75–82. [CrossRef]
60. Neelima, A.; Singh, K.M. Perceptual Hash Function based on Scale-Invariant Feature Transform and Singular Value Decomposition. *Comput. J.* **2018**, *59*, 1275–1281. [CrossRef]
61. Kozat, S.S.; Venkatesan, R.; Mihcak, M.K. Robust perceptual image hashing via matrix invariants. In Proceedings of the 2004 International Conference on Image Processing (ICIP), Singapore, 24–27 October 2004; pp. 3443–3446.
62. Ding, K.; Meng, F.; Liu, Y.; Xu, N.; Chen, W. Perceptual Hashing Based Forensics Scheme for the Integrity Authentication of High Resolution Remote Sensing Image. *Information* **2018**, *9*, 229. [CrossRef]
63. Xia, G.S.; Hu, J.W.; Hu, F.; Shi, B.G.; Bai, X.; Zhong, Y.F.; Zhang, L.P. AID: A Benchmark Dataset for Performance Evaluation of Aerial Scene Classification. *Ieee Trans. Geo. Remote Sens.* **2017**, *55*, 3965–3981. [CrossRef]
64. Xia, G.S.; Bai, X.; Ding, J.; Zhu, Z.; Belongie, S.; Luo, J.; Zhang, L. DOTA: A large-scale dataset for object detection in aerial images. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, USA, 18–22 June 2018; pp. 3974–3983.
65. Yang, Y.; Newsam, S. Bag-of-visual-words and spatial extensions for land-use classification. In Proceedings of the International Conference on Advances in Geographic Information Systems, San Jose, CA, USA, 2–5 November 2010; pp. 270–279.
66. Zhu, Q.Q.; Zhong, Y.F.; Zhao, B.; Xia, G.S.; Zhang, L.P. Bag-of-Visual-Words Scene Classifier With Local and Global Features for High Spatial Resolution Remote Sensing Imagery. *IEEE Geo. Remote Sens. Lett.* **2016**, *13*, 747–751. [CrossRef]
67. Dai, D.; Yang, W. Satellite Image Classification via Two-Layer Sparse Coding With Biased Image Representation. *IEEE Geo Remote Sens. Lett.* **2011**, *8*, 173–176. [CrossRef]
68. Montavon, G.; Samek, W.; Müller, K.R. Methods for interpreting and understanding deep neural networks. *Digital Signal Process.* **2018**, *73*, 1–15. [CrossRef]
69. Xiong, H.K.; Gao, X.; Li, S.H.; Xu, Y.H.; Wang, Y.Z.; Yu, H.Y.; Liu, X.; Zhang, Y.F. Interpretable, structured and multimodal deep neural networks. *Recogn Artif. Intell.* **2018**, *31*, 1–11.