Computers, Materials & Continua

Tech Science Press

DOI: 10.32604/cmc.2023.039540

ARTICLE

Image Steganalysis Based on Deep Content Features Clustering

Chengyu Mo[1,2], Fenlin Liu[1,2], Ma Zhu[1,2,*], Gengcong Yan[3], Baojun Qi[1,2] and Chunfang Yang[1,2]

[1]Henan Provincial Key Laboratory of Cyberspace Situational Awareness, Zhengzhou, 450001, China

[2]Zhengzhou Science and Technology Institute, Zhengzhou, 450001, China

[3]School of Science, Aalto University, Espoo, 02150, Finland

*Corresponding Author: Ma Zhu. Email: qingling800@163.com

Received: 03 February 2023 Accepted: 12 April 2023 Published: 08 October 2023

ABSTRACT

The training images with obviously different contents to the detected images will make the steganalysis model perform poorly in deep steganalysis. The existing methods try to reduce this effect by discarding some features related to image contents. Inevitably, this should lose much helpful information and cause low detection accuracy. This paper proposes an image steganalysis method based on deep content features clustering to solve this problem. Firstly, the wavelet transform is used to remove the high-frequency noise of the image, and the deep convolutional neural network is used to extract the content features of the low-frequency information of the image. Then, the extracted features are clustered to obtain the corresponding class labels to achieve sample pre-classification. Finally, the steganalysis network is trained separately using samples in each subclass to achieve more reliable steganalysis. We experimented on publicly available combined datasets of Bossbase1.01, Bows2, and ALASKA#2 with a quality factor of 75. The accuracy of our proposed pre-classification scheme can improve the detection accuracy by 4.84% for Joint Photographic Experts Group UNIversal WAvelet Relative Distortion (J-UNIWARD) at the payload of 0.4 bits per non-zero alternating current discrete cosine transform coefficient (bpnzAC). Furthermore, at the payload of 0.2 bpnzAC, the improvement effect is minimal but also reaches 1.39%. Compared with the previous steganalysis based on deep learning, this method considers the differences between the training contents. It selects the proper detector for the image to be detected. Experimental results show that the pre-classification scheme can effectively obtain image subclasses with certain similarities and better ensure the consistency of training and testing images. The above measures reduce the impact of sample content inconsistency on the steganalysis network and improve the accuracy of steganalysis.

KEYWORDS

Steganalysis; deep learning; pre-classification

1 Introduction

Digital steganography is a technique that embeds secret information in the redundancy of multimedia data such as digital images, video, audio, and text to achieve covert communication. During the past more than 20 years, researchers have proposed a series of image steganography

algorithms, including early classical techniques such as Least Significant Bit Replace (LSBR) [1], Least Significant Bit Match (LSBM) [2], JSteg [3], F5 [4] and adaptive steganography algorithms that comply with the architecture of "distortion cost function + syndrome-trellis codes (STCs)," such as Highly Undetectable steGO (HUGO) [5], Spatial UNIversal WAvelet Relative Distortion (S-UNIWARD) [6], J-UNIWARD [6], Tong's method [7], Correlational Multivariate Gaussian (CMG) [8]. Accordingly, researchers have proposed many steganalysis approaches, such as early classical Chi-square Attack [9], block effect detection [10], and histogram estimation detection [11]. There are also subsequent high-dimensional feature detection methods such as Spatial Rich Model (SRM) [12], Discrete Cosine Transform Residual (DCTR) [13], Gabor Filter Residual (GFR) [14] and features combinations method [15]. In recent years, inspired by the excellent performance of deep learning in image classification, researchers have also introduced deep learning into steganalysis and proposed many excellent approaches.

A well-designed preprocessing layer can be essential in related research [16]. Most existing steganalysis methods based on deep learning first preprocess the images to obtain high-frequency signals with rich steganographic noise and perform operations such as convolution, regularization, and the activation function to extract features and detect the stego images. According to whether the kernels of the preprocessing layer are learnable, the existing methods can be divided into deep steganalysis-based deterministic preprocessing and deep steganalysis-based learnable preprocessing.

The kernels of the preprocessing layer in deep steganalysis-based deterministic preprocessing are fixed, and their parameters no longer participate in the backpropagation of training after initialization. Qian et al. [17] proposed the Gaussian-Neuron Convolutional Neural Network (GNCNN), which uses a fixed 5×5 high-pass filter kernel in the preprocessing layer to eliminate image content interference and enhance the steganography signal. Then a Convolutional Neural Network (CNN) equipped with a Gaussian activation function detects the images with hidden information. Xu et al. [18] also used the high-pass filter kernel of the GNCNN network for preprocessing in their method. Inspired by SRM features, they take the absolute activation (ABS) layers, batch-normalization (BN) layers, and TanH activation function to effectively capture the symbolic symmetry of residual information and limit the range of feature values. Zeng et al. [19] utilized 25 fixed Discrete Cosine Transform (DCT) kernels for preprocessing and applied multiple subnetworks to form the information after quantization and truncation operations for steganalysis. Subsequently, Zeng et al. [20] improved the model in [19] by taking three parallel subnetworks to realize the steganalysis of large-scale Joint Photographic Experts Group (JPEG) images. Li et al. [21] proposed the ReLU Sigmoid and TanH Network (ReStNet), which uses the linear filter, the nonlinear filters in SRM, and the Gabor filter to preprocess the image, then respectively feeds three preprocessing results to three subnetworks with different activation functions for steganalysis. The Joint Photographic Experts Group Convolutional Neural Network (JPEGCNN) proposed by Gan et al. [22] uses a preprocessing layer with a size of 3×3 to capture neighborhood pixel correlation better, then extracts features by stacking convolution-activation-pooling operations and adopting dropout to improve the performance and generalization ability.

The kernels of the preprocessing layer in deep steganalysis-based learnable preprocessing are learnable and can be automatically updated and optimized in the subsequent training backpropagation after initialization. In 2014, Tan et al. [23] proposed the TanNet, which extracts feature maps with a three-stage stacked convolutional autoencoder, then takes a fully connected neural network to detect the stego images. This method initializes the first layer with the SRM filter kernel. Ye et al. [24] directly initialized the preprocessing layer using 30 residual filter kernels in the SRM model. Then,

the truncated linear unit (TLU) activation function in the first layer was used to learn the distribution of steganographic information. Additionally, they introduced channel selection information [25] to detect stego images. In the Steganalysis Residual Network (SRNet) [26], an end-to-end steganalysis network proposed by Boroumand et al., all filter kernels are randomly initialized and continuously updated in subsequent training, and a shortcut structure [27] is also adopted to increase the diversity of steganalysis features. Zeng et al. proposed the Wider SEparate-then-Reunion Network (WISERNet) [28], which uses the SRM high-pass filter kernel to initialize a layer of convolution of each channel, then updates the convolution kernels to learn the features from different channels of the color images. The Element Wise Network (EWNet) proposed by Su et al. [29] randomly initializes the preprocessing layer, which can continuously learn and optimize in network training. Moreover, its fully convolutional structure avoids excessive loss of steganographic information and realizes the steganalysis of JPEG images with different sizes.

The above methods mainly focus on preprocessing input images and designing convolutional neural network structures. Some of them have exceeded the steganalysis methods based on rich models. Those methods indicate that deep learning methods have become the mainstream of current steganography detection research. However, using a large number of labeled data-driven deep learning steganalysis methods cannot be separated from training and testing consistent data. Current methods do not differ in processing noisy images with complex textures and smooth images with simple content. When designing the above deep steganalysis methods, the consistency of training images and images to be detected should have been considered. The previous research results show that the performance of steganalysis based on deep learning tends to deteriorate when the texture complexity, statistical distribution, and subject content between test images and training images are inconsistent [30–32]. For this problem, Pibre et al. [30] found that the steganalysis based on a convolutional neural network in the Clairvoyant scenario has a specific generalization ability for different datasets. However, this scenario's particularity leads to the application's limitation in real-world applications. Zhang et al. [31] and Zhang et al. [32] tweaked the feature extraction network to extract features less affected by image content. Although their methods can reduce the high false alarm rate caused by apparent differences in image content, they also discard many features related to image content, which can characterize the difference caused by steganography. Therefore, the detection accuracy has been negatively affected. The technology proposed by Abukhodair et al. [33] selects optimal features and effectively classifies big data. This method effectively reduces computational time and increases the accuracy of classification. In the traditional field of steganalysis, Amirkhani et al. [34] proposed a steganalysis framework based on image content pre-classification, which pre-classifies training samples based on non-zero DCT coefficient ratios. A classifier is trained specifically to improve the performance of steganography detection based on low-dimensional detection features. Li et al. [35] proposed a "clustering and classification" JPEG steganalysis method, which classifies training and test samples based on the horizontal and vertical intra-block co-occurrence matrices of the absolute values of the DCT coefficients and improves the detection ability. Lu et al. [36] proposed a steganalysis framework based on pre-classification and feature selection, which utilizes the relationship between adjacent image data to pre-classify samples. The improvement of detection performance is verified in steganalysis based on high-dimensional steganalysis features.

Inspired by the operation of pre-classification of image samples in traditional steganalysis [34–36], this paper proposes a deep-learning steganalysis model that uses image content information to cluster samples to solve the above-mentioned problem. This method improves the extraction

method of classification features by directly using convolutional neural networks to extract content classification features, avoiding the domain knowledge required for the manual design of features. First, wavelet decomposition is performed on the images to obtain low-frequency information. Then the convolutional neural network model extracts features that can describe the image content. Next, we cluster features to obtain pre-classification of the samples based on image content. Finally, the steganalysis model is trained individually with each sub-class of data. The proposed method maintains the sample consistency between the training and the testing phase and improves the reliability of steganalysis.

2 Problem Description

Compared to traditional steganography, adaptive steganography algorithms change pixels or coefficients in regions that are difficult to model and detect. The current typical adaptive steganography algorithms follow the architecture "distortion cost function + STCs." First, a distortion cost function ρ measures the detectability of changing a pixel or coefficient. Then, the secret information is encoded into a stego sequence with minimum overall distortion, viz.

$$S = arg \min_{S} \sum_{i=1}^{N} \rho\left(c_i, s_i\right) \tag{1}$$

where $C = \{c_1, c_2, \cdots, c_N\}$ denotes a sequence of cover pixels or coefficients. $S = \{s_1, s_2, \cdots, s_N\}$ denotes a sequence of stego pixels or coefficients. $\rho\left(c_i, s_i\right)$ denotes the distortion caused by changing the cover pixel or coefficient c_i to s_i. Due to the vast storage and time overhead required to search for minimum-distortion sequences from all possible stego sequences, the STCs were used first by Pevný et al. [5] to encode the secret message into an approximate minimum-distortion sequence of stego pixels or coefficients in 2010. After that, the STCs are still the dominant method adopted by adaptive steganography, although some improved coding methods have been proposed one after another. And currently, steganography researchers mainly focus on the design of better distortion cost functions.

After more than a decade of efforts, researchers have successively designed many distortion cost functions with excellent performance. Although distortion cost functions have their characteristics, almost all of them have the common feature that the distortion is usually more minor when changing the pixels or coefficients in the regions with more obvious color changes or more complex textures. Therefore, most of the pixels or coefficients changed during the steganography cluster in these regions. Since the color of the edges of different content objects in the image varies significantly, the texture of the areas containing a large number of small-sized objects is complex. The distribution of pixels or coefficients changed during steganography is closely related to the image content. Taking the three images shown in row 1 of Fig. 1 selected from the Bossbase 1.01 as an example, we used J-UNIWARD to embed random information into them with an embedding ratio of 0.4 bpnzAC. By comparing the change position maps given in row 2 of Fig. 1, it can be found that the modifications in the first two images mainly concentrate on the edges of the flowers, and the distribution of the changed positions is relatively similar. However, there are differences in the embedding areas of each image. In the third image, the modifications mainly concentrate on the edges of buildings, people, and the complex areas of texture containing many small windows and doors. The distribution of the altered positions is significantly different from the previous two images.

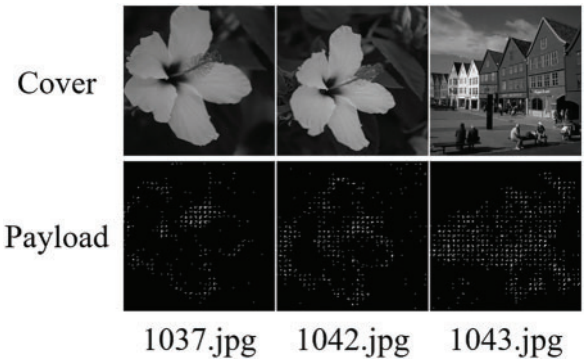

Figure 1: The positions of the coefficients that were changed when J-UNIWARD was used

Four datasets, the Face Recognition Technology (FERET) [37], Oxford 102 Flowers [38], Kaggle[1] flower, and Stanford Dogs [39] were used to test the performance of existing steganalysis methods based on deep learning when the contents of the train data and the test data do not match. The FERET dataset consists of face images. The Oxford 102 Flowers and Kaggle flower datasets both consisted of flower images and were merged into the dataset Flowers. The Stanford Dogs dataset consists of images with the content of dogs. First, 10,000 images were randomly selected from the FERET, Flower, and Stanford Dogs datasets. Then the images were cropped into a square image starting from the top left corner according to the shortest edge cropping principle. Each cropped image was saved as a grayscale JPEG image with a size of 256×256 and a quality factor (QF) of 75 by the resize operation in the python Pillow library. The cover training set, validation set, and testing set were randomly selected from each set of 10,000 grayscale images in the ratio of 4:1:5, respectively. Then, the J-UNIWARD algorithm was used to embed random information into each image with payload 0.4 bpnzAC to generate the corresponding stego image. Finally, the EWNet [29] was trained with the selected training and validation sets. The models trained with three datasets, FERET, Flower, and Stanford Dogs, were abbreviated as EWNet_FERET, EWNet_Flower, and EWNet_Dogs, respectively. Table 1 gives the detection accuracy of the three models for different classes of testing sets. For the FERET testing set, the accuracy of EWNet_FERET is 11.68% higher than the accuracy of EWNet_Flower. The accuracy is 5.39% higher than that of EWNet_Dogs. Similar results can be found for the Flower and Stanford Dogs testing sets. So the models trained with training datasets consistent with the detected objects significantly outperform those trained with inconsistent datasets.

Table 1: The detection accuracy of EWNet in the case of cover source mismatch

TRN\TST on	FERET	Flower	Dogs
FERET	**95.24%**	82.98%	67.43%
Flower	83.56%	**89.84%**	73.51%
Dogs	89.85%	84.77%	**78.59%**

In summary, the inconsistency between the contents of the training samples and the object to be detected significantly impacts on the performance of steganalysis based on deep learning. The

[1]The dataset can be downloaded from here: https://www.kaggle.com/datasets/alxmamaev/flowers-recognition

performance of the deep steganalysis model trained with training samples inconsistent with the object to be detected has significant shortcomings.

3 Method

It is universally acknowledged that deep neural networks have shown excellent performance in image content classification. We proposed a steganalysis method based on image content deep clustering to address the degradation of steganalysis performance caused by the image content inconsistency between the training samples and the object to be detected. The basic idea of the method is to cluster the training images into sub-classes based on the deep features expressing the image contents and train a particular steganalysis network for each sub-class. In this way, the consistency between the training samples and the object to be detected is achieved, and the performance of steganalysis is improved.

As shown in Fig. 2, in the training phase, the image content is first separated from the noise to reduce the impact of noise, including steganographic noise, on the image content feature extraction. Secondly, the deep convolutional neural network with excellent performance in image content classification is used to extract deep features from the training images after noise removal for distinguishing images based on contents. Then, the training samples are clustered according to the extracted features, viz., the training images with close content are divided into the same sub-class. Moreover, the corresponding deep steganalysis network is trained for each sub-class of the sample. In the detecting stage, the image content is first separated from the noise, and the deep convolutional neural network is used to extract the deep features. Then, according to the clustering results of the training stage, the input images are classified to determine the appropriate deep steganalysis network. Finally, the determined deep steganalysis network is used to detect whether the image contains hidden information.

The key of the proposed method lies in noise interference separation and image clustering based on deep features. Therefore, these two parts are described in detail below.

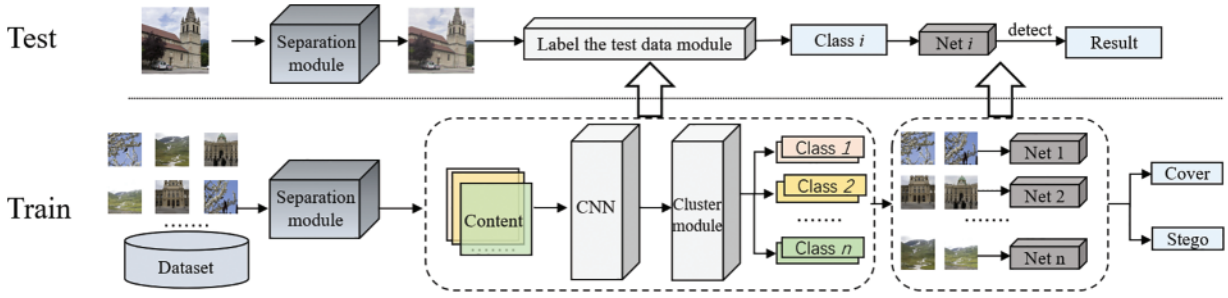

Figure 2: Overall framework of the proposed steganalysis method

3.1 Noise Interference Separation

When the image is transformed into a frequency domain representation, the content information of the image mainly concentrates on the low-frequency components. Correspondingly, the color change information of the edges and textures of the image content object is mainly reflected by high-frequency components. Adaptive steganography embeds information by changing the pixels or coefficients in the edges of image-content objects and areas with more complex textures. That means the steganographic noise is mainly added to the high-frequency components of the image. However,

adaptive steganography has minimal impact on image content. The results of existing adversarial samples show that even the slightest interference applied to the image may lead to misjudgment in the image content classification [40]. To avoid the inconsistency between the image to be detected and the training sample caused by this misjudgment as much as possible, Daubechies wavelets, which perform excellently in noise removal, are used to separate the image noise.

Firstly, a first-level Daubechies wavelet decomposition is performed on the image to obtain its low-frequency component LL, horizontal component LH, vertical component HL, and diagonal component HH. Fig. 3 shows the low-frequency component obtained by the first-level Daubechies wavelet decomposition with different vanishing moments on an image, of which coefficients are mapped to [0, 255]. It can be seen that during the wavelet decomposition, as the disappearance moment is larger, the energy after decomposition is more concentrated, and the image content presented by the low-frequency components becomes clearer. Nevertheless, the larger vanishing moment results in a greater computational cost of the wavelet decomposition. Therefore, according to the value of a vanishing moment usually set in steganalysis, the Daubechies wavelets with the vanishing moment of 8 were selected to decompose the image. The obtained low-frequency components are used as input for the next step of image-deep clustering or classification.

Figure 3: First-stage Daubechies wavelet low-frequency components at different vanishing moments

3.2 Image Clustering Based on Deep Content Features

To solve the problem of steganalysis performance degradation caused by the content inconsistency between training images and images to be detected, we proposed the idea of dividing the training images into multiple sub-classes according to the image content. The corresponding steganalysis network is trained with each subclass of training images. The complicacy of image contents makes it difficult to manually determine in advance which images are close enough to fit into the same class. Therefore, the images are often clustered according to some features, and those with similar features are clustered into the same class. However, traditional image clustering methods often use the differences between some traditional hand-designed image features to measure whether the images are similar. In the existing research, the features extracted based on deep neural networks perform better in image content recognition than traditional hand-designed image features.

Activated by the above view, we cluster the training image based on deep features, as shown in Fig. 4. In the clustering method, the fully connected layer and Softmax are removed from the classical deep neural network VGG16 (seeing in Fig. 5) in the current image content recognition. The remaining backbone is used as an image content feature extractor. The image content in-depth features are extracted from the low-frequency component LL after performing the first-level Daubechies wavelet decomposition for each training image. Then, the classical clustering algorithm is used to cluster the

content-deep features extracted from the low-frequency components of the training images into k training image subsets C_1,C_2,...,C_k. Then, each subset of cover images obtained by clustering and its corresponding stego images are used to train the corresponding steganalysis network. In the detecting phase, the sub-class of the input image is distinguished based on the distance between the content-deep features of the input image and the center of the content-deep feature in each sub-class of training image so that the most appropriate steganography detector is determined.

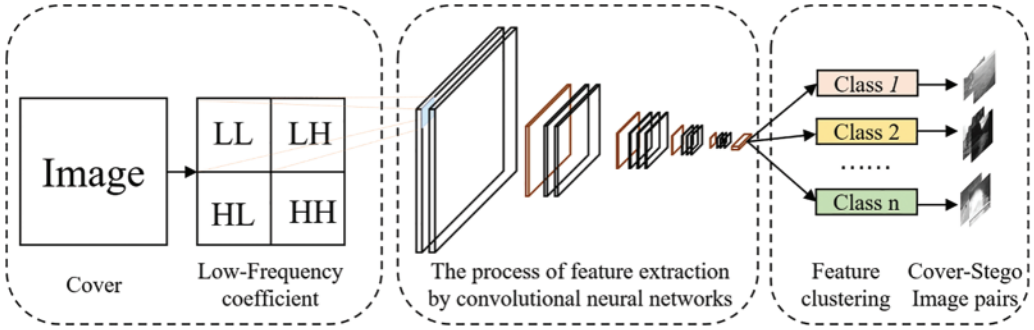

Figure 4: Clustering method of training images

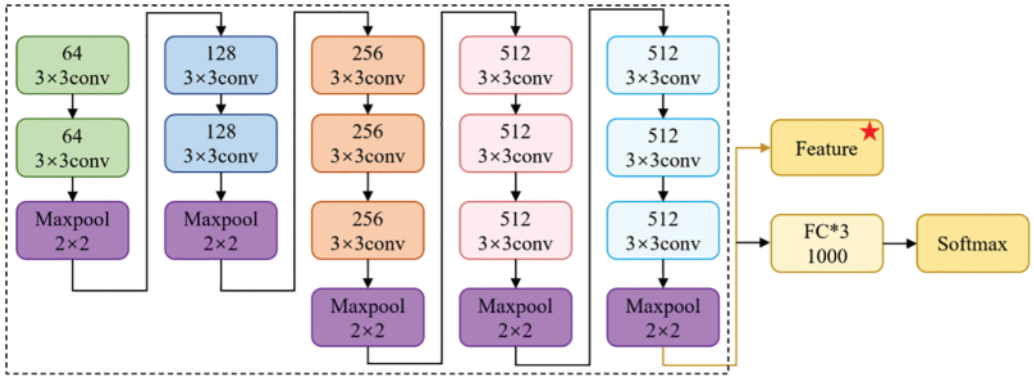

Figure 5: VGG16 convolutional neural network structure and the star-marked output represents the extracted image content features

4 Experimental Results and Analysis

4.1 Datasets and Experimental Environment

Datasets: The images used in the experiments were generated from three publicly available image libraries: Bossbase 1.01, Bows2, and ALASKA #2. In the following text, unless otherwise specified, Bossbase refers to Bossbase 1.01, Bows refers to Bows2, and ALASKA refers to ALASKA #2. Each image dataset Bossbase and Bows, contains 10,000 grayscale images with a size of 512×512. And the dataset ALASKA contains 80,005 JPEG images with QF 75. Images in the three image datasets were stored as grayscale JPEG images with a size of 256×256 and a quality factor of 75. Then, the J-UNIWARD steganography algorithm was used to embed pseudo-random information with payloads of 0.1, 0.2, 0.3, and 0.4 bpnzAC in all three cover image datasets, and 12 sets of corresponding stego images were obtained.

Model training: Considering the size of the dataset and the convergence of network training, we set the number of sub-classes of image clusters as 2 and 4, respectively. After clustering, for each sub-class of training images, the EWNet was trained with 90,000 iterations, an initial learning rate of 0.001and an adjusted learning rate of 0.0001 after the first 50000 iterations. The GPU used in model training was NVIDIA GeForce GTX 1080Ti.

Hyperparameter optimization: The network was trained by the mini-batch stochastic gradient descent (SGD) optimizer Adamax with $\beta_1 = 0.91$, $\beta_2 = 0.999$ and $\in = 1 \times e^{-8}$. The batch size was set to 32 (16 cover-stego pairs). The convolutional layers were initialized with the normal distribution initializer with a standard deviation of 0.01, and $2 \times e^{-4}$ L2 regularization was used to alleviate overfitting. The convolutional layer was set with no bias. The parameter of the batch normalization layer was learnable with a decay rate of 0.9. The ReLU activation function is used for nonlinear processing.

4.2 Detection Performance of Submodels

We tested the performance of the steganalysis submodel trained with each sub-class of images after clustering. First, the 10,000 cover images in Bossbase were equally divided into one group of training cover images and one group of testing cover images. According to the principle of one-to-one correspondence with the cover image, stego images were also divided into one group of training stego images and one group of testing stego images. Then, the training cover images were clustered into two sub-classes, Bossbase_C0 and Bossbase_C1, by the method proposed in this paper. The number of images in each class is shown in Table 2. Each sub-class of training cover images and its corresponding stego images were used to train the corresponding deep steganalysis models EWNet_BC0 and EWNet_BC1, respectively. Finally, the class center obtained by clustering was used to classify the test images to determine the proper detection model for steganalysis.

Table 2: The number of cover images in each sub-class after clustering the 5000 training cover images in the Bossbase dataset

Bossbase	Bossbase_C0	Bossbase_C1	Total
	3196	1804	5000

To compare the steganalysis performance before and after clustering, we randomly selected 3196 cover images from the training cover images to form an image group Bossbase_R0. For each payload, the selected cover images in Bossbase_R0 and the corresponding stego images were used to train the corresponding detection model, which detected the 3142 pairs of test images classified into sub-class Bossbase_C0. 1804 cover images were randomly selected from the training cover images to form an image group Bossbase_R1. For each payload, the selected cover images in Bossbase_R1 and the corresponding stego images were used to train the corresponding detection model, which was used to detect the 1858 pairs of test images classified into sub-class Bossbase_C1. The specific training and test dataset partitioning scheme is shown in Fig. 6.

Table 3 shows the detection accuracy of the submodel trained by each sub-classes of training cover and the corresponding stego images at each embedding ratio. From the experimental results, the detection accuracy of each steganalysis submodel trained by clustered images is higher than that of the steganalysis model trained by randomly selected images. The improvements are maximum at 0.4 bpnzAC. Especially, the accuracy of the steganalysis submodel EWNet_BC1 trained by cover

images in Bossbase_C1 is higher than that of the model training by cover images in Bossbase_R1 by more than 3%. However, the accuracies of the steganalysis submodels trained by cover images in Bossbase_C0 exceed that of the steganalysis submodels trained by cover images in Bossbase_R0 by a small margin. The reason may be that the images in Bossbase_C0 account for 63.8% of the total training cover images, so there is a lot of overlap between Bossbase_C0 and Bossbase_R0.

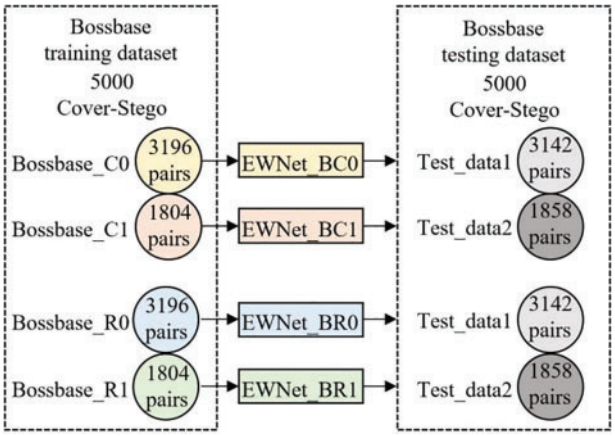

Figure 6: Bossbase dataset experimental data division scheme

Table 3: The detection accuracy of the submodel trained by each sub-class of images after clustering

Payload\Class	Bossbase_C0	Bossbase_R0	Bossbase_C1	Bossbase_R1
0.4	**89.47%**	89.16%	**83.69%**	80.27%
0.3	**82.38%**	82.11%	**75.24%**	73.33%
0.2	**73.35%**	73.12%	**67.41%**	65.74%
0.1	**62.38%**	62.17%	**58.64%**	57.45%

To eliminate the performance difference caused by the different number of training images, three cover image datasets were merged into one cover image dataset, referred to as the cover BBA dataset (Bossbase_Bows_ALASKA). And the corresponding stego image dataset is referred to as the stego BBA dataset. 90,005 images were randomly selected from BBA as training cover images, and the remaining 10,000 images were used as test cover images. 90,005 training cover images were first clustered into four sub-classes: BBA_C0, BBA_C1, BBA_C2, and BBA_C3. The number of images in each sub-class is shown in Table 4. 5000 cover images were randomly selected from each sub-class. Then, four cover image sets, BBA_C0_5K, BBA_C1_5K, BBA_C2_5K, and BBA_C3_5K, were obtained. For each payload, the stego images corresponding to 5000 cover images were selected to form a group of training stego images. After training 90 epochs, the corresponding steganalysis submodel was obtained. During steganalysis, 5000 test cover images were randomly selected from 10,000 test cover images. The class centers obtained during clustering were used to classify the selected test cover images and the corresponding test stego images, then determined the corresponding steganalysis submodels detect them. The number of test image pairs categorized into each sub-class is shown in Fig. 7.

To test the performance of the steganalysis submodels before and after clustering, 5000 training cover images were randomly selected from the 90,005 training cover images to form an image set BBA_R_5K. For each payload, the cover images in the BBA_R_5K and their corresponding stego images were used to train the corresponding steganalysis submodel, which was used to detect above 5000 test cover images and their corresponding stego images.

Table 4: The amount of pre-classified training data in the BBA dataset

BBA	BBA_C0	BBA_C1	BBA_C2	BBA_C3	Total
	18623	19836	23343	28203	90005

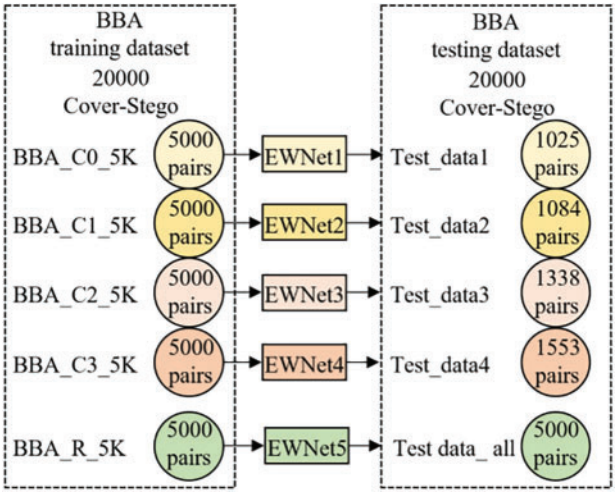

Figure 7: Experimental data partitioning scheme for the same number of training subsets

Table 5 shows the detection accuracy of different submodels for the test images with different payloads. The detection accuracy of each steganalysis submodel trained with clustered images is significantly higher than that of the submodel trained by the randomly selected images in most cases when the numbers of training images are equal. In particular, when the payload is 0.4, the accuracy of the steganalysis submodel trained with the BBA_C1_5K image set is improved by about 7.5%. The accuracy of the steganalysis submodel trained with the BBA_C2_5K image set is improved by the minimum margin, nearly 1%. The experimental results show that under the same number of training samples, this scheme can also overcome the interference caused by differences in image content to a certain extent, especially at a high embedding rate. Through comparative analysis of images, it is found that the content of images in BBA_C1_5K is more similar, while BBA_C2_5K is more complex, so the improvement effect of BBA_C1_5K is most obvious.

4.3 Detection Performance of the Ensemble Model

To test the overall steganalysis performance of the proposed method, we trained the steganalysis submodels by using the four types of training cover images obtained by clustering in the previous section BBA_C0, BBA_C1, BBA_C2, BBA_C3, and their corresponding stego images. The submodels were combined into an ensemble detector EWNet_Cluster. The remaining 10,000 test cover images

and corresponding test stego images were classified by using the class center obtained by clustering to determine the corresponding detection submodel and detect them. To compare the steganalysis performance before and after clustering, all the training cover images and their corresponding stego images were directly used to train a steganalysis model EWNet_All and then used to detect 10,000 test cover images and their corresponding test stego images. At the same time, we also use the method proposed in [36] to pre-classify the samples. When training, we also select K as 4 and train each submodel to obtain the ensemble model EWNet_ Lu. The specific training and test dataset partitioning scheme is shown in Fig. 8.

Table 5: The detection accuracy of the submodels trained by the same number of images selected from the BBA dataset

Payload\Class	BBA_C0_5K	BBA_C1_5K	BBA_C2_5K	BBA_C3_5K	BBA_R_5K
0.4	**83.46%**	**84.13%**	**77.58%**	**82.39%**	76.63%
0.3	**74.10%**	**78%**	71.41%	73.25%	73.47%
0.2	**67.61%**	**71.36%**	64.20%	**66.58%**	65.25%
0.1	**59.90%**	**60.01%**	56.58%	**58.15%**	57.24%

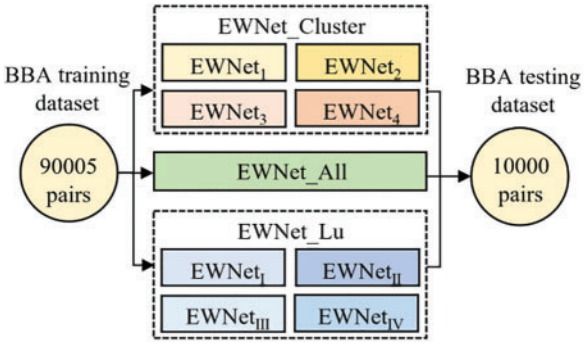

Figure 8: Experimental scheme under the same total number of training sets

Table 6 shows the detection accuracy of the ensemble model at different payloads. Compared with the detection model trained by all data, the detection accuracy of the ensemble model EWNet_Cluster trained with clustered images significantly exceeds that of the model EWNet_All and EWNet_Lu. Especially when the payload is 0.4, the detection accuracy of the EWNet_Cluster is about 4.84% higher than that of EWNet_All and about 2.56% higher than that of EWNet_Lu. Compared to EWNet_All, when the payload is 0.2, the accuracy of the model EWNet_Cluster improved the least, but also by 1.39%. At the same embedding rate, the accuracy rate of our solution increased by 3.16% compared to EWNet_Lu. In the case of an ensemble model, this scheme can appropriately partition the different content samples, making the model focus more on the steganographic signal rather than the content information of the image.

To view the detection accuracy of each submodel in detail, Table 7 reports the accuracy of each submodel separately. Under different embedding ratios, compared with the steganalysis model EWNet_All trained with all training images, the steganalysis submodels' accuracy with BBA_C0, BBA_C1, and BBA_C3 images are higher than that of EWNet_All. In particular, the detection accuracy of the submodel trained with BBA_C1 has the most significant improvement, which reaches

8.84% at payload 0.4. Submodels trained with only BBA_C2 images have slightly lower detection accuracy than EWNet_All at a payload not larger than 0.3. This should be attributed to the fewer training images and complex image content. The experimental results show that the detection accuracy of the model can be significantly improved under different embedding rates. However, as the embedding rate decreases, BBA_ C2 will slightly decrease because the texture of this subclass of samples is more complex and challenging to detect. The scheme is valid, as in EWNet_ All measures the average of all accuracy rates, and some difficult-to-detect class results will be lower than the average.

Table 6: The detection accuracy of EWNet_Cluster, EWNet_All and EWNet_Lu trained by BBA dataset

Payload\Net	EWNet_Cluster	EWNet_All	EWNet_Lu
0.4	**81.56%**↑	76.72%	79.00%
0.3	**76.44%**↑	72.96%	72.64%
0.2	**68.24%**↑	66.85%	65.08%
0.1	**59.26%**↑	57.73%	57.36%

Table 7: The detection accuracy of each submodel in EWNet_Cluster

Payload\Class	BBA_C0	BBA_C1	BBA_C2	BBA_C3
0.4	80.54%	85.56%	77.63%	82.62%
0.3	75.70%	79.70%	72.49%	77.87%
0.2	70.32%	71.39%	64.06%	68.15%
0.1	58.92%	61.13%	57.03%	59.76%

The above experimental results show that under the same data set, the ensemble detector can effectively improve the accuracy of steganalysis. It shows that the clustering based on a deep content features scheme can overcome the problem of poor detection performance caused by the content mismatch between the training images and the detected images to a certain extent and improve the detection accuracy.

5 Conclusion

In this paper, an image steganalysis method based on deep content features clustering is proposed. The powerful learning ability of the convolutional neural network is used to extract image content features which are used to cluster the training images. Therefore, a special deep steganalysis submodel is obtained for each class of training images, and an ensemble detection model is formed by combining all sub-models. During detection, the most suitable steganalysis submodel is found based on the deep content features extracted from the input image to minimize the data difference between the training images and the input image. Experimental results show that compared with the model trained by all training images, the proposed method can significantly improve the accuracy of steganalysis based on convolutional neural networks. Compared to randomly extracting subsets from the dataset for training, in the submodel detection performance comparison experiment, the performance can improve by more than 3% on a single subclass of Bossbase at most. The effect can also be improved

by up to 7.5% with the same number of subsets on the BBA dataset. The overall performance of steganalysis can be improved by up to 4.84% when the payload is 0.4.

This method is just an attempt to apply image content clustering based on deep learning features to steganalysis. How to cluster the training images should be determined by the impact of clustering on steganalysis performance. This is also one of the directions to be further explored.

Acknowledgement: The authors are grateful to the anonymous reviewers for their constructive comments and suggestions.

Funding Statement: This work is supported by the National Natural Science Foundation of China (Nos. 61872448, 62172435, 62072057), the Science and Technology Research Project of Henan Province in China (No. 222102210075).

Author Contributions: study conception and design: Chengyu Mo, Fenlin Liu; data collection: Ma Zhu, Gengcong Yan; analysis and interpretation of results: Chengyu Mo, Ma Zhu, Fenlin Liu; draft manuscript preparation: Chengyu Mo, Baojun Qi, Chunfang Yang. All authors reviewed the results and approved the final version of the manuscript.

Availability of Data and Materials: In the study, we used the Bossbase1.01, Bows2, and ALASKA#2 datasets, which are publicly available and can be accessed via the citation links in the paper.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] C. Kurak and J. McHugh, "A cautionary note on image downgrading," in *Proc. of ACSAC*, San Antonio, Texas, USA, pp. 153–159, 1992.
[2] T. Sharp, "An implementation of key-based digital signal steganography," in *Proc. of IHW*, Pittsburgh, PA, USA, pp. 13–26, 2001.
[3] D. Upham, *JPEG-Jsteg-v4*, 1993. [Online]. Available: https://zooid.org/~paul/crypto/jsteg/
[4] W. Andreas, "F5-A steganographic algorithm," in *Proc. of IHW*, Pittsburgh, PA, USA, pp. 289–302, 2001.
[5] T. Pevný, T. Filler and P. Bas, "Using high-dimensional image models to perform highly undetectable steganography," in *Proc. of IH*, Calgary, AB, Canada, pp. 161–177, 2010.
[6] V. Holub and J. Fridrich, "Digital image steganography using universal distortion," in *Proc. of IH&MMSec*, Montpellier, France, pp. 59–68, 2013.
[7] Y. C. Tong, J. Q. Ni, W. K. Su and X. L. Hu, "Spatial image steganography incorporating adjacent," in *Proc. of ICAIS*, Qinghai, China, pp. 412–423, 2022.
[8] J. Q. Ni, Y. C. Tong, X. L. Hu, W. K. Su and X. G. Kang, "Spatial image steganography using a correlational model," in *Proc. of ICAIS*, Qinghai, China, pp. 400–411, 2022.
[9] A. Westfeld and A. Pfitzmann, "Attacks on steganographic systems," in *Proc. of IHW*, Dresden, Germany, pp. 61–76, 1999.
[10] J. Fridrich, M. Goljan and D. Hogea, "Attacking the outguess," in *Proc. of MM&Sec*, Juan-les-Pins, French Riviera, pp. 3–6, 2002.
[11] J. Fridrich, M. Goljan and D. Hogea, "Steganalysis of JPEG images: Breaking the F5 algorithm," in *Proc. of IH*, Noordwijkerhout, Netherlands, pp. 310–323, 2002.
[12] J. Fridrich and J. Kodovský, "Rich models for steganalysis of digital images," *IEEE Transactions on Information Forensics and Security*, vol. 7, no. 3, pp. 868–882, 2012.

[13] V. Holub and J. Fridrich, "Low complexity features for JPEG steganalysis using undecimated DCT," *IEEE Transactions on Information Forensics and Security*, vol. 10, no. 2, pp. 219–228, 2015.
[14] X. F. Song, F. L. Liu, C. F. Yang and Y. Zhang, "Steganalysis of adaptive JPEG steganography using 2D gabor filters," in *Proc. of IH&MMSec*, Portland, OR, USA, pp. 15–23, 2015.
[15] Z. C. Wang, Z. X. Qian, X. P. Zhang and S. Li, "An improved steganalysis method using feature combinations," in *Proc. of ICAIS*, New York, NY, USA, pp. 115–127, 2019.
[16] T. F. Yang, J. Wu, G. R. Feng, X. Chang and L. H. Liu, "A deep learning approach to detection of warping forgery in images," in *Proc. of ICAIS*, Hohhot, China, pp. 109–118, 2020.
[17] Y. L. Qian and J. Dong, "Deep learning for steganalysis via convolutional neural networks," in *Proc. of MWSF*, San Francisco, CA, USA, pp. 171–180, 2015.
[18] G. S. Xu, H. Z. Wu and Y. Q. Shi, "Structural design of convolutional neural networks for steganalysis," *IEEE Signal Processing Letters*, vol. 23, no. 5, pp. 708–712, 2016.
[19] J. S. Zeng, S. Q. Tan, B. Li and J. W. Huang, "Pre-training via fitting deep neural network to rich-model features extraction procedure and its effect on deep learning for steganalysis," in *Proc. of MWSF*, Burlingame, CA, USA, pp. 44–49, 2015.
[20] J. S. Zeng, S. Q. Tan, B. Li and J. W. Huang, "Large-scale JPEG image steganalysis using hybrid deep-learning framework," *IEEE Transactions on Information Forensics and Security*, vol. 13, no. 5, pp. 1200–1214, 2018.
[21] B. Li, W. H. Wei, A. Ferreira and S. Q. Tan, "ReST-Net: Diverse activation modules and parallel subnets-based CNN for spatial image steganalysis," *IEEE Signal Processing Letters*, vol. 25, no. 5, pp. 650–654, 2018.
[22] L. Gan, J. J. Chen, Y. L. Chen, Z. J. Jin and W. X. Han, "JPEGCNN: A transform domain steganalysis model based on convolutional neural network," in *Proc. of ICAIS*, New York, NY, USA, pp. 566–577, 2019.
[23] S. Q. Tan and B. Li, "Stacked convolutional auto-encoders for steganalysis of digital images," in *Proc. of APSIPA*, Chiang Mai, Thailand, pp. 1–4, 2014.
[24] J. Ye, J. Q. Ni and Y. Yi, "Deep learning hierarchical representations for image steganalysis," *IEEE Transactions on Information Forensics and Security*, vol. 12, no. 11, pp. 2545–2557, 2017.
[25] T. Denemark, V. Sedighi, V. Holub, R. Cogranne and J. Fridrich, "Selection-channel-aware rich model for steganalysis of digital images," in *Proc. of WIFS*, Atlanta, GA, USA, pp. 48–53, 2014.
[26] M. Boroumand, M. Chen and J. Fridrich, "Deep residual network for steganalysis of digital images," *IEEE Transactions on Information Forensics and Security*, vol. 14, no. 5, pp. 1181–1193, 2019.
[27] K. M. He, X. Y. Zhang, S. Q. Ren and J. Sun, "Deep residual learning for image recognition," in *Proc. of CVPR*, Las Vegas, NV, USA, pp. 770–778, 2016.
[28] J. S. Zeng, S. Q. Tan, G. Q. Liu, B. Li and J. W. Huang, "WISERNet: Wider separate-then-reunion network for steganalysis of color images," *IEEE Transactions on Information Forensics and Security*, vol. 14, no. 10, pp. 2735–2748, 2019.
[29] A. T. Su, X. F. Zhao and X. L. He, "Arbitrary-sized JPEG steganalysis based on fully convolutional network," in *Proc. of IWDW*, Beijing, China, pp. 197–211, 2021.
[30] L. Pibre, J. Pasquet, D. Ienco and M. Chaumont, "Deep learning is a good steganalysis tool when embedding key is reused for different images, even if there is a cover source-mismatch," in *Proc. of MWSF*, San Francisco, California, USA, pp. 1–11, 2016.
[31] X. P. Zhang, X. W. Kong, P. D. Wang and B. Wang, "Cover-source mismatch in deep spatial steganalysis," in *Proc. of IWDW*, Chengdu, China, pp. 71–83, 2019.
[32] L. Zhang, S. Abdullahi, P. He and H. X. Wang, "Dataset mismatched steganalysis using subdomain adaptation with guiding feature," *Telecommunication Systems*, vol. 80, no. 2, pp. 263–276, 2022.
[33] F. Abukhodair, W. Alsaggaf, A. T. Jamal, S. Abdel-Khalek and R. F. Mansou, "An intelligent metaheuristic binary pigeon optimization-based feature selection and big data classification in a MapReduce environment," *Mathematics*, vol. 9, no. 20, pp. 2627, 2021.
[34] H. Amirkhani and M. Rahmati, "New framework for using image contents in blind steganalysis systems," *Journal of Electronic Imaging*, vol. 20, no. 1, pp. 013016, 2011.

[35] W. X. Li, T. Zhang, G. Hu and K. Xie, "Image pre-classification to improve accuracy of universal steganalysis," in *Proc. of ICSESS*, Beijing, China, pp. 364–368, 2014.
[36] J. C. Lu, G. Zhou, C. F. Yang, Z. Y. Li and M. J. Lan, "Steganalysis of content-adaptive steganography based on massive datasets pre-classification and feature selection," *IEEE Access*, vol. 7, pp. 21702–21711, 2019.
[37] P. Phillips, H. Moon and S. Rizvi, "The FERET evaluation methodology for face-recognition algorithms," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 10, pp. 1090–1104, 2022.
[38] M. Nilsback and A. Zisserman, "Automated flower classification over a large number of classes," in *Proc. of ICVGIP*, Bhubaneswar, India, USA, pp. 722–729, 2008.
[39] A. Khosla, N. Jayadevaprakash, B. Yao and F. F. Li, "Novel dataset for fine-grained image categorization: Stanford dogs," in *CVPR Workshop on Fine-Grained Visual Categorization (FGVC)*, vol. 2, no. 1, Citeseer, 2011.
[40] J. Y. Liu, W. M. Zhang, D. D. Hou, Y. J. Liu, H. Y. Zha *et al.,* "Detection based defense against adversarial examples from the steganalysis point of view," in *Proc. of CVPR*, Long Beach, CA, USA, pp. 4825–4834, 2019.